

```
<!DOCTYPE html>
<html>

<head>
  <title>JavaScript DOM Project</title>
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.2/css/all.min.css"
integrity="sha512-HK5fgLBL+yu6dm/Ii3z4xhLSUyZgTT9tuc/hSrtw6uzJOvgRr2a9jyxxTlELY+B+xFAM
JKVSTbpM/CuL7qxO8w=="
    crossorigin="anonymous" />
  <style>
    @import
url('https://fonts.googleapis.com/css2?family=Hanalei&family=Roboto+Slab&display=swap'
);
    @import
url('https://fonts.googleapis.com/css2?family=Shadows+Into+Light&display=swap');

    * {
      box-sizing: border-box;
    }

    body {
      font-family: 'Roboto Slab', serif;
    }

    .box {
      font-family: 'Hanalei', cursive;
      position: relative;
      width: 100px;
      height: 100px;
      border: 1px solid black;
      top: 0;
      left: 0;
      font-size: 3em;
      line-height: 100px;
      text-align: center;
    }

    .wheel {
      position: absolute;
      left: 0px;
```

```
    top: 0px;
}

.gameArea {
    position: absolute;
    left: 0px;
    top: 80px;
    width: 300px;
    height: 103px;
    border: 1px solid #333;
    overflow: hidden;
}

.topMessage {
    position: absolute;
    left: 50%;
    top: 10px;
    background-color: black;
    width: 80%;
    height: 60px;
    color: white;
    padding: 5px;
    font-size: 2em;
    font-family: 'Shadows Into Light', cursive;
    transform: translate(-50%, 0);
    text-align: center;
    line-height: 40px;
}

.message {
    text-align: center;
    font-size: 1em;
    color: white;
}

.btn {
    margin: auto;
    width: 80%;
    display: block;
    padding: 10px;
```

```

        color: white;
        font-size: 1.5em;
        border-radius: 25px;
    }

    .myDiv {
        padding: 20px;
        height: 200px;
        background-color: dimgrey;
        width: 50%;
        margin: 190px auto;
    }
</style>

</head>

<body>
    <div class="myDiv">Hello World</div>
    <script src="apps3.js"></script>
</body>

</html>

const output = document.querySelector('.myDiv');
const iconImages = ['<i class="fas fa-heart"></i>', '<i class="fas fa-dice-d20"></i>',
    '<i class="far fa-life-ring"></i>', '<i class="fas fa-apple-alt"></i>', '<i class="fas
fa-lemon"></i>', '<i class="fas fa-pepper-hot"></i>'];
output.innerHTML = '';
const messageTop = makerElement(output, 'div', 'JavaScript Slot Machine',
'topMessage');
const gameArea = makerElement(output, 'div', '', 'gameArea');
const btn = makerElement(output, 'button', 'SPIN', 'btn');
const message = makerElement(output, 'div', 'message', 'message');
const game = {
    total: 5,
    inPlay: false,
    coins: 100,
    speed: 15,
    totItems: iconImages.length,
    main: []
};
let spinner = 500;

```

```

window.addEventListener('DOMContentLoaded', init);
btn.addEventListener('click', (e) => {
  if (btn.textContent == 'SPIN' && !game.inPlay) {
    btn.textContent = 'STOP';
    btn.style.backgroundColor = 'red';
    spinner = 500;
    startSpin();
  } else {
    stopGamePlay();
  }
})

function init() {
  ///console.log('ready');
  btn.style.backgroundColor = 'green';
  gameArea.style.width = game.total * 100 + 'px';
  let leftPos = (document.body.clientWidth - (game.total * 100)) / 2;
  ///console.log(leftPos);
  gameArea.style.left = leftPos + 'px';
  for (let i = 0; i < game.total; i++) {
    game.main[i] = makerElement(gameArea, 'div', '', 'wheel')
    for (let x = 0; x < game.totItems; x++) {
      const el = makerElement(game.main[i], 'div', iconImages[x], 'box');
      let myColor = x > 2 ? 'red' : 'blue';
      if (x == 0) {
        myColor = 'purple';
      }
      if (x > 4) {
        myColor = 'green';
      }
      if (x == 4) {
        myColor = 'orange';
      }
      el.style.color = myColor;
      el.faceValue = x + 1;
    }
    game.main[i].style.left = i * 100 + 'px';
  }
}

function makerElement(parent, ele, html, myClass) {
  const el = document.createElement(ele);

```

```

    el.classList.add(myClass);
    el.innerHTML = html;
    parent.append(el);
    return el;
}

function updateMessage(html) {
    message.innerHTML = html;
}

function startSpin() {
    game.coins--;
    updateMessage(`You have ${game.coins} left.`);
    game.inPlay = true;
    spinner = 500;
    ///console.log('Spinning ' + game.inPlay);
    for (let i = 0; i < game.total; i++) {
        game.main[i].mover = Math.floor(Math.random() * 150) + 10;
    }
    game.ani = requestAnimationFrame(spin);
}

function spin() {
    spinner--;
    if (spinner <= 0) {
        stopGamePlay();
    }
    let holder = [];
    for (let i = 0; i < game.total; i++) {
        let el = game.main[i];
        let elY = el.offsetTop;
        if (el.mover > 0) {
            el.mover--;
            elY += game.speed;
            if (elY > -150) {
                elY -= 100;
                const last = el.lastElementChild;
                el.prepend(last);
            }
            if (el.mover == 0 && elY % 50 != 0) {
                el.mover++;
            }
        }
    }
}

```

```

        el.style.top = elY + 'px';
    } else {
        let viewEl = el.children[2];
        let outputVal = elY == -200 ? viewEl.faceValue : '-';
        let tempObj = {
            'txt': viewEl.faceValue,
            'elY': elY,
            'outputV': outputVal,
            'output': viewEl.textContent
        }
        holder.push(tempObj);
    }
}

if (holder.length >= game.total) {
    stopGamePlay();
    holder.sort();
    console.log(holder);
    const myObj = {};
    holder.forEach((val) => {
        if (val.outputV != '-') {
            if (myObj[val.outputV]) {
                myObj[val.outputV]++;
            } else {
                myObj[val.outputV] = 1;
            }
        }
    })
    payout(myObj);
}

if (game.inPlay) {
    game.ani = requestAnimationFrame(spin);
}
}

function payout(score) {
    for (const prop in score) {
        let val = Number(score[prop]); //how many occurrences
        console.log(prop + ' x ' + val);
        let basePay = game.total / 2;
    }
}

```

```
        if (val >= 2) {
            let totalPaid = Math.floor(val * basePay);
            if (prop == '2') {
                console.log('You Got more than 2 - 2s');
                totalPaid *= 5;
            }
            game.coins += totalPaid;
            let html = `Matched item ${prop} X ${val} Payout ${totalPaid} Coins
${game.coins}`;
            updateMessage(html);
        }
    }
}

function stopGamePlay() {
    game.inPlay = false;
    cancelAnimationFrame(game.ani);
    btn.textContent = 'SPIN';
    btn.style.backgroundColor = 'green';
}
```