

게임프로그래밍

이름 : 전상훈
과목 : 게임프로그래밍
날짜 : 11-15

목차

01 원본 코드

02 업그레이드 계획

03 업그레이드 후 코드 소개

04 실행 결과

05 출처

원본 코드

```
<style>canvas { border: 1px solid black; }</style>
<canvas id="mycanvas"/>

<script type="text/javascript">
var canvas = document.getElementById('mycanvas');
canvas.width = 300;
canvas.height = 300;
var context = canvas.getContext('2d');

// 새로운 이미지 객체를 만듦
var img = new Image();
img.onload = function(){
  for (var i = 0; i < 4; i++) {
    for (var j = 0; j < 3; j++) {
      context.drawImage(img, j * 100, i * 50, 100, 50);
    }
  }
}
// 이미지 URL
img.src = 'http://example.com/image/girl-4056684__340.jpg';
</script>
```



업그레이드 계획



Sliding Puzzle

Part 3 업그레이드 후 코드 소개 – 전체 코드

```
<!DOCTYPE html>
<html>

<head>
<style>
  canvas {
    border: 1px solid black;
  }
</style>
<link rel="icon" href="https://github.com/JeonSangHoon0711/slidingpuzzle/blob/main/cat/main.png?raw=true">
</head>

<body onload="setupMainScreen()">
  <canvas id="maincanvas"></canvas>
  <canvas id="rankcanvas"></canvas>
  <audio id="mySound" src="https://github.com/JeonSangHoon0711/slidingpuzzle/raw/main/audio/click.wav"></audio>

  <div id="game-container">
    <canvas id="mycanvas"></canvas>
    <canvas id="scoreboard"></canvas>
  </div>

  <script type="text/javascript">

    var timescore = 0;
    var moviecount = 0;
    var doing = "도전중입니다."

    var gameInterval;
    var timescoreInterval;

    var ranks = [
      { name: '철수', time: 100, moves: 120 },
      { name: '영희', time: 120, moves: 222 },
      { name: '민준', time: 90, moves: 318 }
    ];
    var canvas = document.getElementById('mycanvas');
    canvas.width = 300;
    canvas.height = 300;
    canvas.style.display = 'none';
    var context = canvas.getContext('2d');

    var images = []; // 배열 선언
    images.push({ id: 0, img: new Image(), x: 0, y: 0, width: 100, height: 100 });
    // 새로운 이미지 객체를 만들

    for (let i = 1; i <= 9; i++) {
      var img = new Image();
      img.src = "https://github.com/JeonSangHoon0711/slidingpuzzle/blob/main/cat$(i).png?raw=true";
      images.push({
        id: i, // id 추가
        img: img,
        x: ((i - 1) % 3) * 100,
        y: Math.floor((i - 1) / 3) * 100,
        width: 100,
        height: 100
      });
    }

    function startgame() {
      isGameStarted = true; // 게임 시작
      maincanvas.style.display = 'none'; // 클릭하면 두 번째 화면이 사라지도록 설정
      rankcanvas.style.display = 'none';
      canvas.style.display = 'block'; // 클릭하면 첫 번째 화면이 나타나도록 설정
      scoreboard.style.display = 'block';
      drawImages();
      randswap();
      gameInterval = setInterval(gameend, 100);

      drawScoreBoard();
      starttimeScore();
    }

    function drawImages() {
      context.clearRect(0, 0, canvas.width, canvas.height);
      images.forEach(image => {
        context.drawImage(image.img, image.x, image.y, image.width, image.height);
      });
    }

    function swapImages(i, j) {
      var temp = { x: images[i].x, y: images[i].y };
      images[i].x = images[j].x;
      images[j].x = images[i].x;
      images[i].y = temp.y;
      drawImages();
    }

    function check(i, j) {
      return Math.abs(images[i].x - images[j].x) + Math.abs(images[i].y - images[j].y) === 100;
    }
  </script>
</body>
</html>
```

```
function gameend() {
  var count = 0;
  for (let i = 1; i <= 9; i++) {
    if (images[i].x === ((i - 1) % 3) * 100 && images[i].y === Math.floor((i - 1) / 3) * 100) {
      count++;
    }
  }
  if (count === 9) {
    clearInterval(gameInterval);
    clearInterval(timescoreInterval);
    doing = "만렙중입니다.";
    drawScoreBoard();
    setTimeout(function () {
      alert("성공!");
      var isRecord = confirm("기록을 입력하겠습니까?");
      if (isRecord) {
        // 사용자가 확인을 클릭했을 때의 처리를 여기에 작성합니다.
        var name = prompt("이름을 입력하세요.");
        ranks.push({
          name: name,
          time: timescore,
          moves: moviecount
        });
        drawRank();
      }
      alert("기록이 입력되었습니다.");
      maincanvas.style.display = 'block';
      rankcanvas.style.display = 'block';
      canvas.style.display = 'none';
      scoreboard.style.display = 'none';
      timescore = 0;
      moviecount = 0;
      doing = "도전중입니다.";
    }, 100)
  }
}

function randswap() {
  var v = 0;
  while ((images[0].x != 200 || images[0].y != 200) || v < 500) { // 그림자일 경우까지 지형
    var chose1 = Math.floor(Math.random() * 8) + 1;
    var chose2 = 9;
    v++;
    if (check(chose1, chose2)) {
      swapImages(chose1, chose2);
      drawImages();
    }
  }
}

canvas.addEventListener('click', function (event) {
  var rect = canvas.getBoundingClientRect();
  var x = event.clientX - rect.left;
  var y = event.clientY - rect.top;
  var sound = document.getElementById('mySound');
  sound.currentTime = 0;
  sound.play();
  images.forEach(image => {
    if (x >= image.x && x <= image.x + image.width && y >= image.y && y <= image.y + image.height) {
      switch (image.id) {
        case 8:
          if (check(8, 9)) {
            swapImages(8, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 7:
          if (check(7, 9)) {
            swapImages(7, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 6:
          if (check(6, 9)) {
            swapImages(6, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 5:
          if (check(5, 9)) {
            swapImages(5, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 4:
          if (check(4, 9)) {
            swapImages(4, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 3:
          if (check(3, 9)) {
            swapImages(3, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 2:
          if (check(2, 9)) {
            swapImages(2, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
        case 1:
          if (check(1, 9)) {
            swapImages(1, 9);
            moviecount++;
            drawScoreBoard();
          }
          break;
      }
    }
  });
}
```

```
var scoreboard = document.getElementById('scoreboard');
scoreboard.width = 300;
scoreboard.height = 300;
scoreboard.style.display = 'none';
var container = document.getElementById('game-container');
container.style.display = 'flex';
```

```
var scoreboardContext = scoreboard.getContext('2d'); // 컨텍스트를 가져옴
scoreboardContext.font = '30px Arial'; // 폰트 설정
```

```
function starttimeScore() {
  timescoreInterval = setInterval(function () {
    timescore++;
    drawScoreBoard();
  }, 1000);
}
```

```
function drawScoreBoard() {
  // 스코어 보드를 클리어
  scoreboardContext.clearRect(0, 0, scoreboard.width, scoreboard.height);

  // 첫 번째 행에 timescore와 moviecount 출력
  scoreboardContext.fillText('u2605u2605u2605u2605u2605u2605u2605u2605u2605', 10, 80);
  scoreboardContext.fillText('u2605u2605u2605u2605u2605u2605u2605u2605u2605', 10, 210);
  scoreboardContext.fillText('Time: ' + timescore + ', Moves: ' + moviecount, 10, 120);
}
```

```
// 두 번째 행에 doing 출력
scoreboardContext.fillText(doing, 10, 170);
}
```

```
function setupMainScreen() {
  var maincanvas = document.getElementById('maincanvas');
  maincanvas.width = 600;
  maincanvas.height = 400;
  maincanvas.style.display = 'block';
  var maincontext = maincanvas.getContext('2d');

  // 메인 캔버스에 이미지 출력
  var mainImage = new Image();
  mainImage.src = '/cat/main.png';
  mainImage.onload = function () {
    maincontext.drawImage(mainImage, 0, 0, maincanvas.width, maincanvas.height);
  }
}
```

```
drawRank();
```

```
// 메인 캔버스에 클릭 이벤트 리스너 추가
maincanvas.addEventListener('click', function (event) {
  stangame(), // 클릭하면 stangame 함수 실행
});
```

```
document.addEventListener('keydown', function (event) {
  if (event.key === 'q') { // 'q' 키를 누른 경우와 게임이 시작된 경우
    gameend(); // gameend 함수를 실행
  }
});
```

```
var rankcanvas = document.getElementById('rankcanvas');
rankcanvas.width = 600;
rankcanvas.height = 400;
var rankcontext = rankcanvas.getContext('2d');
```

```
function drawRank() {
  ranks.sort(function (a, b) {
    return a.time - b.time;
  });
  // rankcanvas를 클리어합니다.
  rankcontext.clearRect(0, 0, rankcanvas.width, rankcanvas.height);
  rankcontext.font = '50px Arial';
  rankcontext.fillText('점수판', 205, 60);
}
```

```
rankcontext.font = '20px Arial';
// 각 등수를 rankcanvas에 그립니다.
ranks.forEach(rank, i) => {
  rankcontext.fillText((i + 1) + '위: ' + rank.name + ', 시간: ' + rank.time + ', 움직인 횟수: ' + rank.moves, 10, 30 * (i + 3));
}
```

```
</script>
```

```
</body>
```

```
</html>
```

업그레이드 후 코드 소개

```
<body onload="setupMainScreen()">
  <canvas id="maincanvas"></canvas>
  <canvas id="rankcanvas"></canvas>
  <audio id="mySound" src="https://github.com/JeonSangHoon0711/slidingpuzzle/raw/main/audio/click.wav"></audio>

  <div id="game-container">
    <canvas id="mycanvas"></canvas>
    <canvas id="scoreboard"></canvas>
  </div>
```

업그레이드 후 코드 소개

```
function setupMainScreen() {  
    var maincanvas = document.getElementById('maincanvas');  
    maincanvas.width = 600;  
    maincanvas.height = 400;  
    maincanvas.style.display = 'block';  
    var maincontext = maincanvas.getContext('2d');  
  
    // 메인 캔버스에 이미지 출력  
    var mainImage = new Image();  
    mainImage.src = './cat/main.png';  
    mainImage.onload = function () {  
        maincontext.drawImage(mainImage, 0, 0, maincanvas.width, maincanvas.height);  
    }  
  
    drawRank();  
  
    // 메인 캔버스에 클릭 이벤트 리스너 추가  
    maincanvas.addEventListener('click', function (event) {  
        startgame(); // 클릭하면 startgame 함수 실행  
    });  
}
```



업그레이드 후 코드 소개

```
function startgame() {  
    isGameStarted = true; // 게임 시작  
    maincanvas.style.display = 'none'; // 클릭하면 두 번째 화면이 사라지도록 설정  
    rankcanvas.style.display = 'none';  
    canvas.style.display = 'block'; // 클릭하면 첫 번째 화면이 나타나도록 설정  
    scoreboard.style.display = 'block';  
    drawImages();  
    randswap();  
    gameInterval = setInterval(gameend, 100);  
  
    drawScoreBoard()  
    starttimescore();  
}
```


업그레이드 후 코드 소개

```
var canvas = document.getElementById('mycanvas');
canvas.width = 300;
canvas.height = 300;
canvas.style.display = 'none';
var context = canvas.getContext('2d');

var images = []; // 배열 선언
images.push({ id: 0, img: new Image(), x: 0, y: 0, width: 100, height: 100 });
// 새로운 이미지 객체를 만듦

for (let i = 1; i <= 9; i++) {
var img = new Image();
img.src = `https://github.com/JeonSangHoon0711/slidingpuzzle/blob/main/cat/cat${i}.png?raw=true`;
images.push({
  id: i, // id 추가
  img: img,
  x: ((i - 1) % 3) * 100,
  y: Math.floor((i - 1) / 3) * 100,
  width: 100,
  height: 100
});
}
```



업그레이드 후 코드 소개

```
function drawImages() {  
  context.clearRect(0, 0, canvas.width, canvas.height);  
  images.forEach(image => {  
    context.drawImage(image.img, image.x, image.y, image.width, image.height);  
  });  
}
```



업그레이드 후 코드 소개

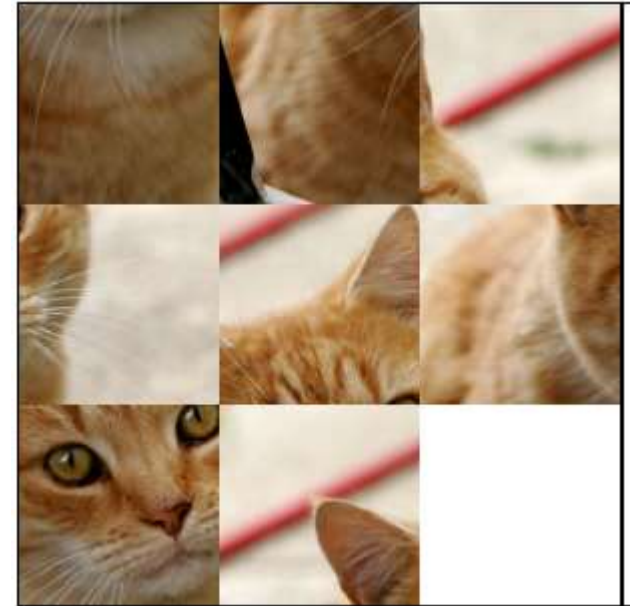
```
function swapImages(i, j) {  
    var temp = { x: images[i].x, y: images[i].y };  
    images[i].x = images[j].x;  
    images[i].y = images[j].y;  
    images[j].x = temp.x;  
    images[j].y = temp.y;  
    drawImages();  
}
```



```
function check(i, j) {  
    return Math.abs(images[i].x - images[j].x) + Math.abs(images[i].y - images[j].y) === 100;  
}
```

업그레이드 후 코드 소개

```
function randswap() {  
    var v = 0;  
    while ((images[9].x != 200 || images[9].y != 200) || v < 500) { // 그림타일 랜덤위치 지정  
        var chose1 = Math.floor(Math.random() * 8) + 1;  
        var chose2 = 9;  
        v++;  
  
        if (check(chose1, chose2)) {  
            swapImages(chose1, chose2);  
            drawImages();  
        }  
    }  
}
```



업그레이드 후 코드 소개

```

canvas.addEventListener('click', function (event) {
    var rect = canvas.getBoundingClientRect();
    var x = event.clientX - rect.left;
    var y = event.clientY - rect.top;
    var sound = document.getElementById('mySound');
    sound.currentTime = 0;
    sound.play();
    images.forEach((image, i) => {
        if (x >= image.x && x <= image.x + image.width && y >= image.y && y <= image.y + image.height) {
            switch (image.id) {
                case 8:
                    if (check(8, 9)) {
                        swapImages(8, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 7:
                    if (check(7, 9)) {
                        swapImages(7, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 6:
                    if (check(6, 9)) {
                        swapImages(6, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 5:
                    if (check(5, 9)) {
                        swapImages(5, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 4:
                    if (check(4, 9)) {
                        swapImages(4, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 3:
                    if (check(3, 9)) {
                        swapImages(3, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 2:
                    if (check(2, 9)) {
                        swapImages(2, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
                case 1:
                    if (check(1, 9)) {
                        swapImages(1, 9);
                        moviecount++;
                        drawScoreBoard();
                    }
                    break;
            }
        }
    });
});

```

업그레이드 후 코드 소개

```
function gameend() {
  var count = 0;
  for (let i = 1; i <= 9; i++) {
    if (images[i].x === ((i - 1) % 3) * 100 && images[i].y === Math.floor((i - 1) / 3) * 100) {
      count++;
    }
  }
  if (count === 9) {
    clearInterval(gameInterval);
    clearInterval(timescoreInterval);
    doing = "완성했습니다.";
    drawScoreBoard();
    setTimeout(function () {
      alert("성공");
      var isRecord = confirm("기록을 입력하겠습니까?");
      if (isRecord) {
        // 사용자가 확인을 클릭했을 때의 처리를 여기에 작성합니다.
        var name = prompt("이름을 입력하세요.");
        ranks.push({
          name: name,
          time: timescore,
          moves: moviecount
        });
        drawRank();

        alert("기록이 입력되었습니다.");
        maincanvas.style.display = 'block';
        rankcanvas.style.display = 'block';
        canvas.style.display = 'none';
        scoreboard.style.display = 'none';
        timescore = 0;
        moviecount = 0;
        doing = "도전중입니다.";
      }
    }, 100)
  }
}
```

```
gameInterval = setInterval(gameend, 100);
```

jeonsanghoon0711.github.io 내용:

성공

확인

jeonsanghoon0711.github.io 내용:

기록을 입력하겠습니까?

확인

취소

jeonsanghoon0711.github.io 내용:

이름을 입력하세요.

맹귀

확인

취소

업그레이드 후 코드 소개

```
var scoreboard = document.getElementById('scoreboard');
scoreboard.width = 300;
scoreboard.height = 300;
scoreboard.style.display = 'none';
var container = document.getElementById('game-container');
container.style.display = 'flex';

var scoreboardContext = scoreboard.getContext('2d'); // 컨텍스트를 가져옴
scoreboardContext.font = '30px Arial'; // 폰트 설정

function starttimescore() {
    timescoreInterval = setInterval(function () {
        timescore++;
        drawScoreBoard();
    }, 1000);
}
```

★★★★★★★★

Time: 3, Moves: 0

도전중입니다.

★★★★★★★★

업그레이드 후 코드 소개

```
function drawScoreBoard() {  
    // 스코어 보드를 클리어  
    scoreboardContext.clearRect(0, 0, scoreboard.width, scoreboard.height);  
  
    // 첫 번째 행에 timescore와 moviecount 출력  
    scoreboardContext.fillText('\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605', 10, 80);  
    scoreboardContext.fillText('\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605\u2605', 10, 210);  
    scoreboardContext.fillText('Time: ' + timescore + ', Moves: ' + moviecount, 10, 120);  
  
    // 두 번째 행에 doing 출력  
    scoreboardContext.fillText(doing, 10, 170);  
}
```

★★★★★★★★★★

Time: 3, Moves: 0

도전중입니다.

★★★★★★★★★★

업그레이드 후 코드 소개

```
var rankcanvas = document.getElementById('rankcanvas');  
rankcanvas.width = 600;  
rankcanvas.height = 400;  
var rankcontext = rankcanvas.getContext('2d');
```

점수판

1위: 민준, 시간: 90, 움직인 횟수: 318

2위: 철수, 시간: 100, 움직인 횟수: 120

3위: 영희, 시간: 120, 움직인 횟수: 222

업그레йд 후 코드 소개

```
function drawRank() {  
  ranks.sort(function (a, b) {  
    return a.time - b.time;  
  });  
  // rankcanvas를 클리어합니다.  
  rankcontext.clearRect(0, 0, rankcanvas.width, rankcanvas.height);  
  rankcontext.font = '50px Arial';  
  rankcontext.fillText('점수판', 205, 60);  
  
  rankcontext.font = '20px Arial';  
  // 각 등수를 rankcanvas에 그립니다.  
  ranks.forEach((rank, i) => {  
    rankcontext.fillText((i + 1) + '위: ' + rank.name + ', 시간: ' + rank.time + ', 움직인 횟수: ' + rank.moves, 10, 30 * (i + 3));  
  });  
}
```

점수판

1위: 민준, 시간: 90, 움직인 횟수: 318
2위: 철수, 시간: 100, 움직인 횟수: 120
3위: 영희, 시간: 120, 움직인 횟수: 222

실행 결과

	<p>시간 움직인 횟수</p> <p>★★★★★★★★★★</p> <p>Time: 36, Moves: 82</p> <p>완성했습니다.</p> <p>★★★★★★★★★★</p> <p>클릭시 시작됩니다.</p>
--	---

점수판

1위: 민준, 시간: 90, 움직인 횟수: 318
2위: 철수, 시간: 100, 움직인 횟수: 120
3위: 영희, 시간: 120, 움직인 횟수: 222

출처

https://www.habonyphp.com/2021/01/blog-post_47.html - 원본 소스

<https://www.trudi.com/en/prodotto/puzzle-scorrevole-tigre/>

<https://yusaebyeol.blogspot.com/2020/07/powerpoint-template-free-download-ppt.html>

<https://mixkit.co/free-sound-effects/click/>

<https://www.freeimages.com/kr/photo/cat-s-face-1553769>

ai.com