

[기말과제 의뢰]

1. 소스코드

```
import java.awt.BorderLayout;
import java.awt.Graphics;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import java.util.Random;

import javax.imageio.ImageIO;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;

public class MyImageFrame extends JFrame implements ActionListener {

    private int pieces = 4;
    private int totalPieces = pieces * pieces;
    private int[] pieceNumber;
    private BufferedImage img;

    public MyImageFrame() {
        setTitle("Image Draw Test");
        try {
            img = ImageIO.read(new File("hubble.jpg"));
        } catch (IOException e) {
            System.out.println(e.getMessage());
            System.exit(0);
        }
        pieceNumber = new int[totalPieces];
        for (int i = 0; i < totalPieces; i++) {
            pieceNumber[i] = i;
        }
    }
}
```

```

        add(new MyPanel(), BorderLayout.CENTER);
        JButton button = new JButton("DIVIDE");

        button.addActionListener(this);
        add(button, BorderLayout.SOUTH);
        setSize(img.getWidth(null), img.getHeight(null));
        setVisible(true);
    }

    void divide() {
        Random rand = new Random();
        int ri;
        for (int i = 0; i < totalPieces; i++) {
            ri = rand.nextInt(totalPieces);
            int tmp = pieceNumber[i];

            pieceNumber[i] = pieceNumber[ri];
            pieceNumber[ri] = tmp;
        }
    }

    class MyPanel extends JPanel {
        public void paintComponent(Graphics g) {
            super.paintComponent(g);
            int pieceWidth = img.getWidth(null) / pieces;
            int pieceHeight = img.getHeight(null) / pieces;
            for (int x = 0; x < pieces; x++) {
                int sx = x * pieceWidth;
                for (int y = 0; y < pieces; y++) {
                    int sy = y * pieceHeight;
                    int number = pieceNumber[x * pieces
+ y];

                    int dx = (number / pieces) *
pieceWidth;

                    int dy = (number % pieces) *
pieceHeight;

```

```

g.drawImage(img, dx, dy, dx +
pieceWidth, dy + pieceHeight,
sx, sy, sx +
pieceWidth, sy + pieceHeight, null);
    }
}

}

}

public static void main(String[] args) {

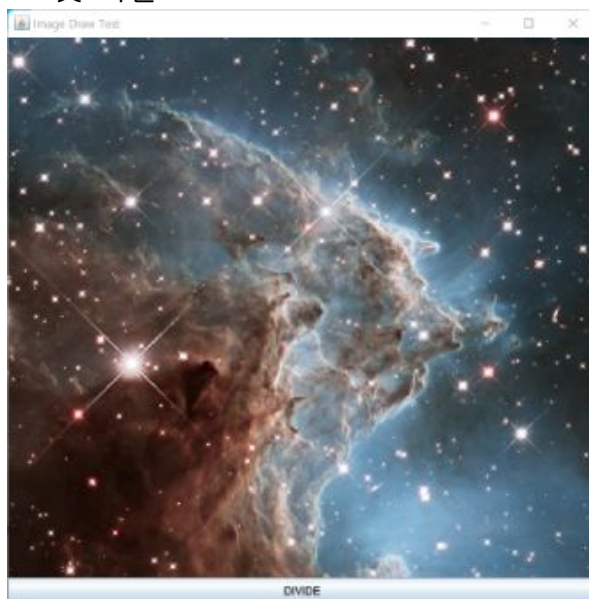
    new MyImageFrame();
}

public void actionPerformed(ActionEvent e) {
    divide();
    repaint();
}
}

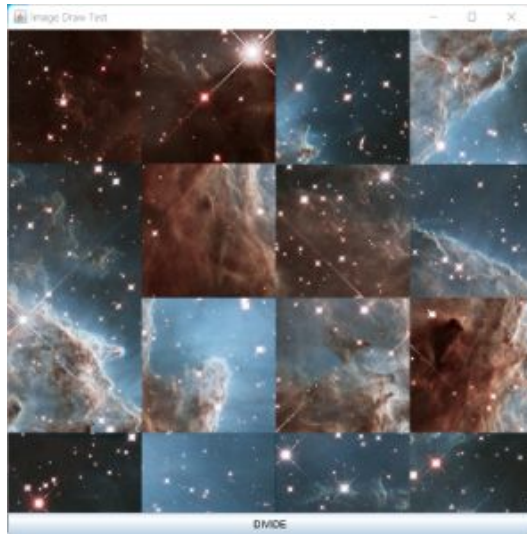
```

3. 실행 결과 화면

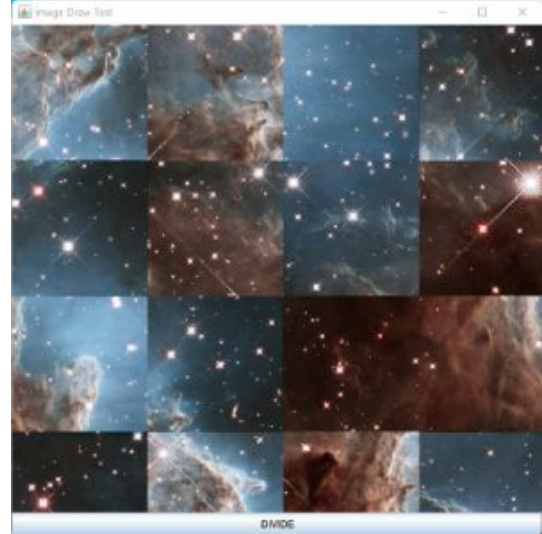
- 첫 화면



- DIVIDE를 처음 눌렀을 때 화면



- DIVIDE를 다시 눌렀을 때 화면



[기획의도 - 기존 프로그램에서 수정/개선하고 싶은 사항]

: 나뉘지고 무작위로 섞인 이미지를 마우스 클릭을 통해서 원래 상태로 바꾸는 프로그램으로 수정 및 개선

- 프로그램 실행시 원하는 파일을 직접 불러오도록 수정.
- 그림 파일을 몇 조각으로 나눌지 직접 입력을 통해 설정하도록 수정.
- 몇 조각으로 나눌지 입력 후 시작 버튼 클릭시 나뉘진 이미지가 보이도록 수정.
- 시작 버튼 클릭과 동시에 타이머로 시간을 재는 기능을 추가 후 상단에 시간이 보여지도록 만듦.
- 하단에 원본 이미지가 보여지도록 만듦.
- 나뉘진 이미지 조각 a를 클릭하고 b를 클릭시 a와 b의 위치가 바뀌도록 만듦.
- 원래 이미지와 동일한 상태로 만들시 걸린 시간과 몇 번의 클릭만에 성공했는지를 보여줌.