벌레를 잡아라!

김가희, 신아영, 진민지 조

목차

1

구현 목표

2

구현 코드

3

구현 결과

구현 목표

Thread를 활용한 게임

JTabbedPane

JPanel

Event

Graphic

JLabel

JFrame

Image

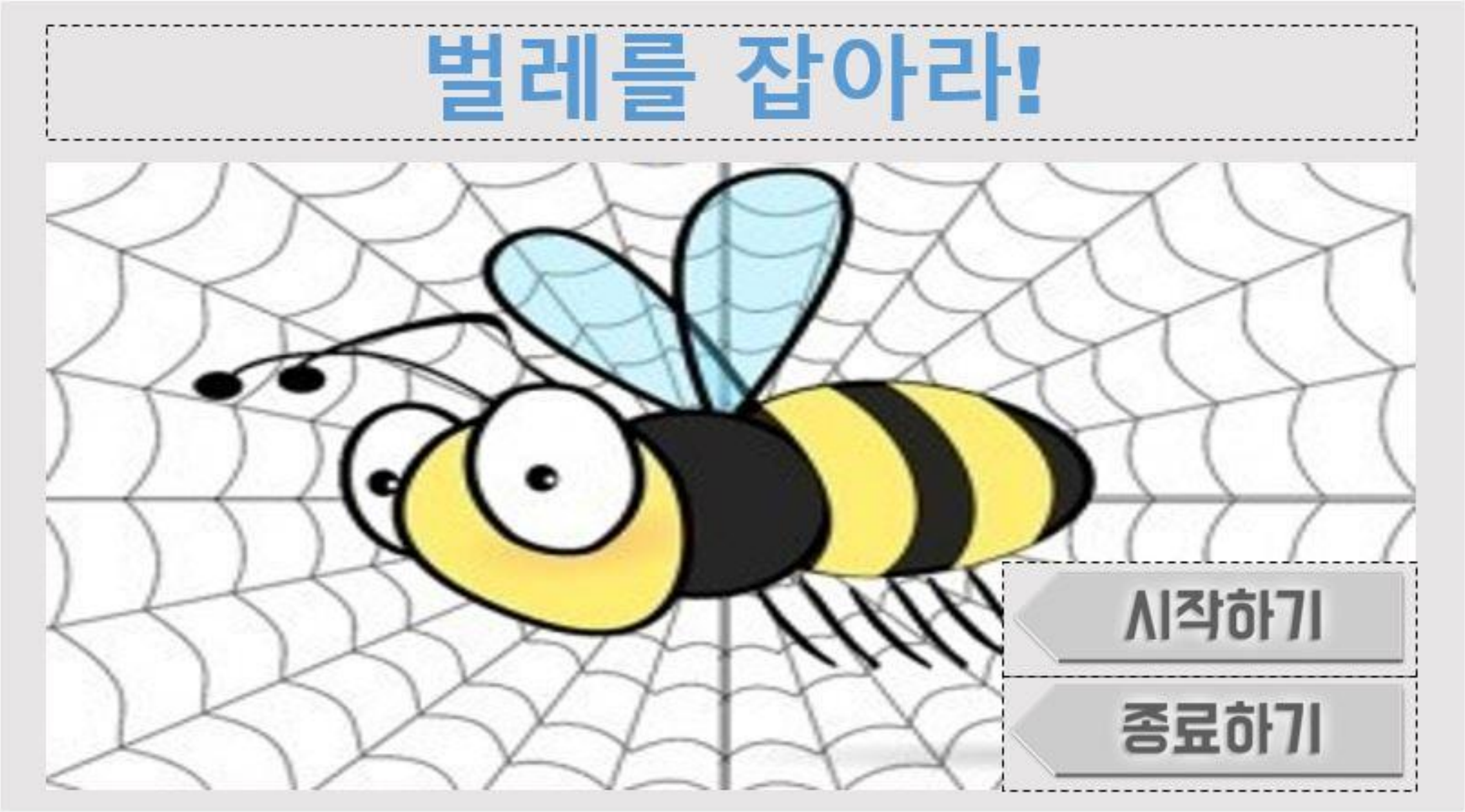
GridLayout

BorderLayout

Textfield

구현 목표 - 레이아웃

1



전체 프레임 - Card Layout

setLayout(card)

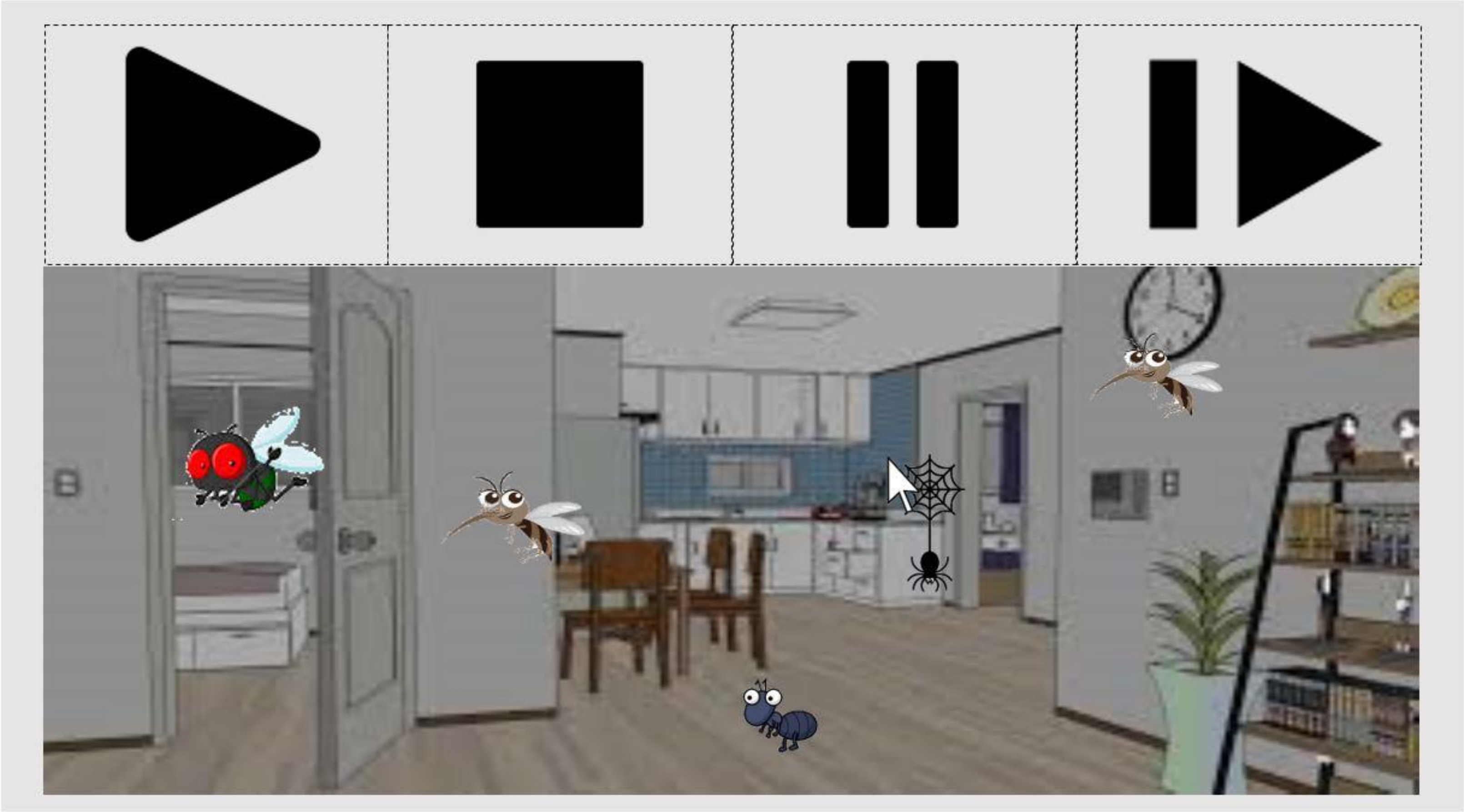
//첫 번째 패널
Jpanel start

paintComponont로
이미지 그리기

이미지 버튼

구현 목표 - 레이아웃

2



//두 번째 패널
centerPanel
paintComponent 배경 넣기

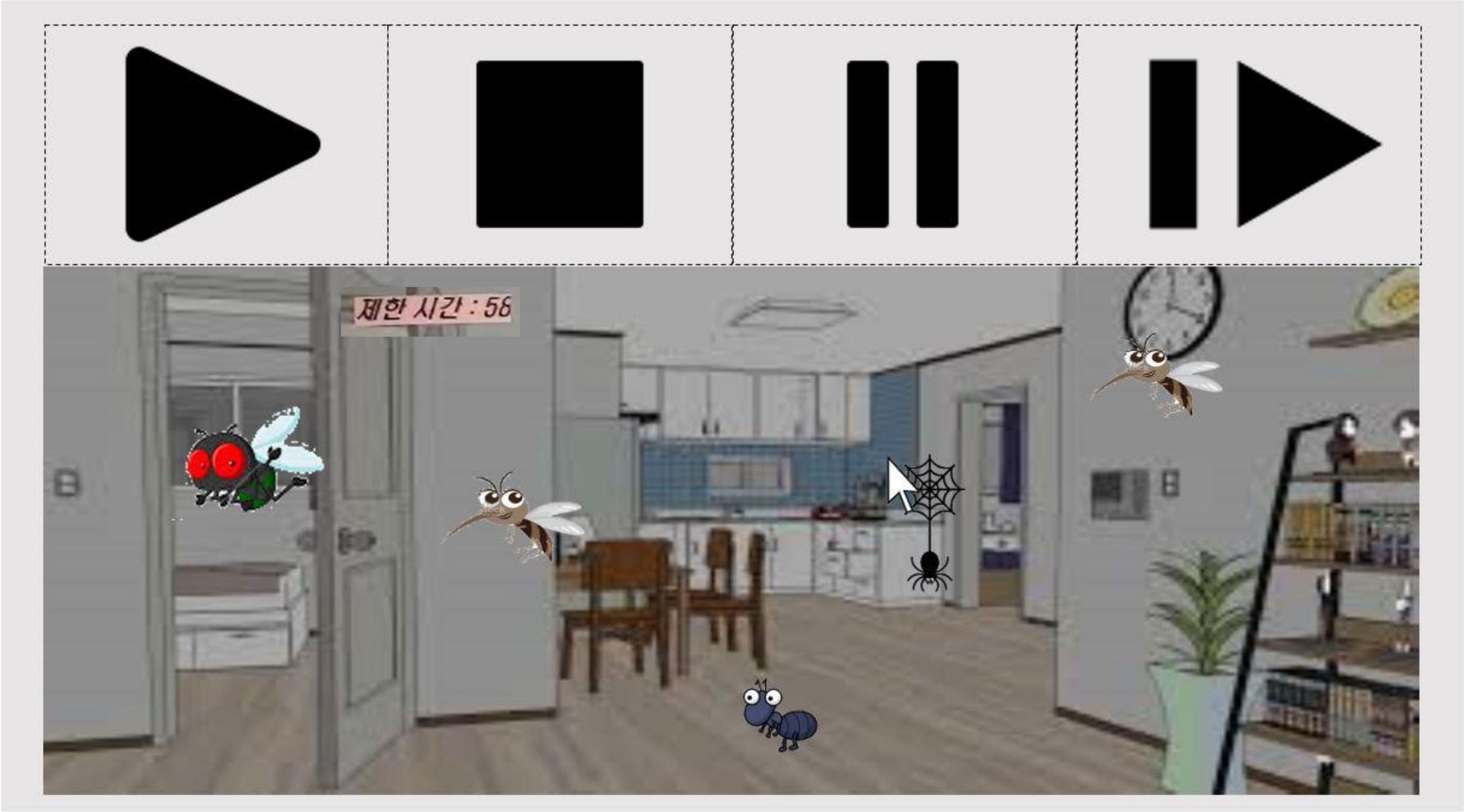
menuPanel
메뉴 버튼 4가지

contentPanel “NORTH”:
menuPanel
contentPanel “CENTER”:
centerPanel

//재생 버튼
centerPanel-bugs[], deathbugs[]
//일시 정지 버튼
thread stop
//되돌아가기 버튼
thread resume

구현 목표 - 레이아웃

2



//타이머 레이블

60초 제한

1초 단위

일시정지 시 쓰레드와 함께 중단

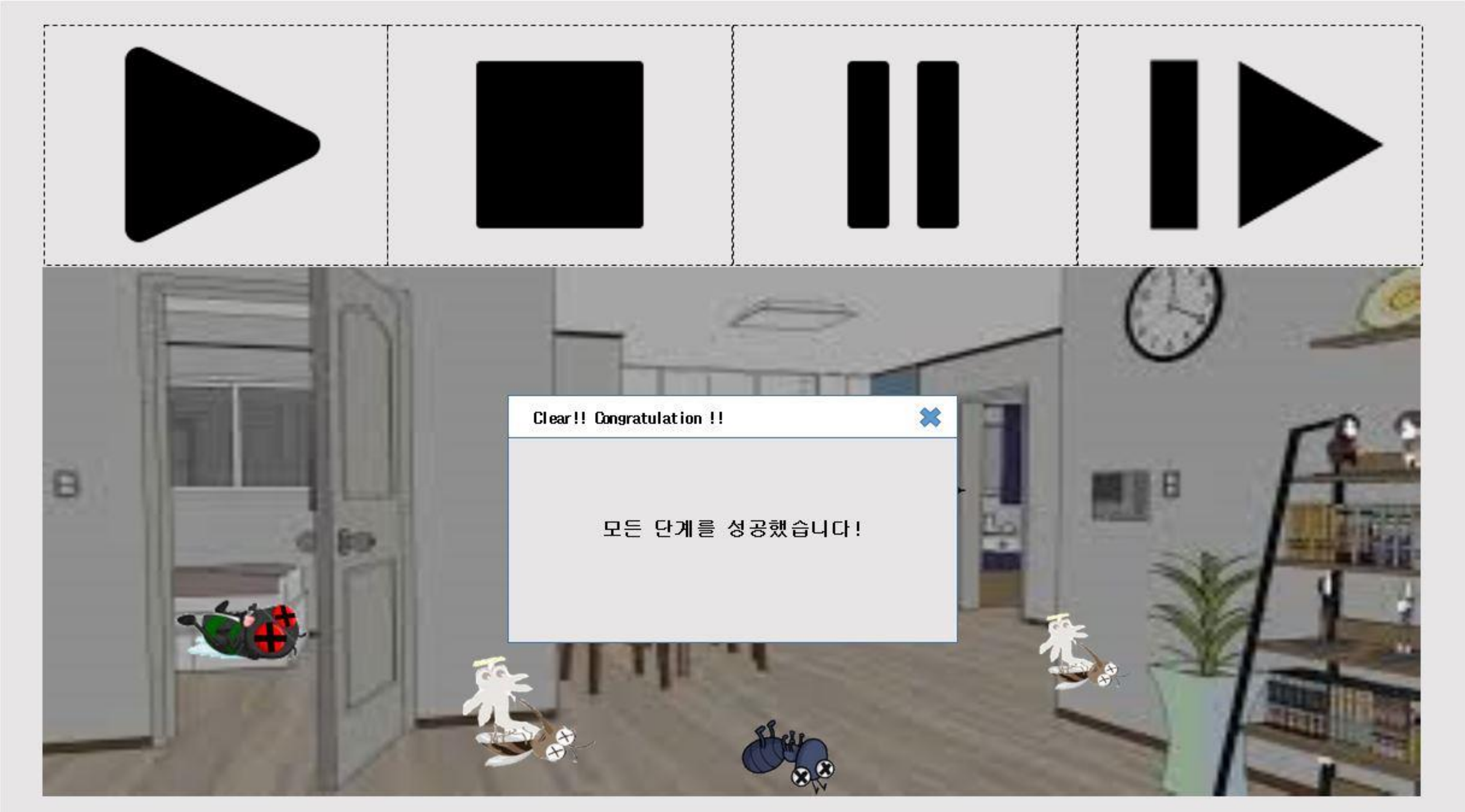
벌레들도 각자 쓰레드로 움직임
쓰레드에 ActionListener추가

time이 0이하일 때 (시간 초과 시)
팝업 창-실패 메시지
fail로 화면 전환

커서 이미지 변경

구현 목표 - 레이아웃

3



모든 단계를 성공했습니다!

모든 레벨 달성 시
팝업 창-성공 메시지

음악 효과 추가

구현 목표 - 레이아웃

4

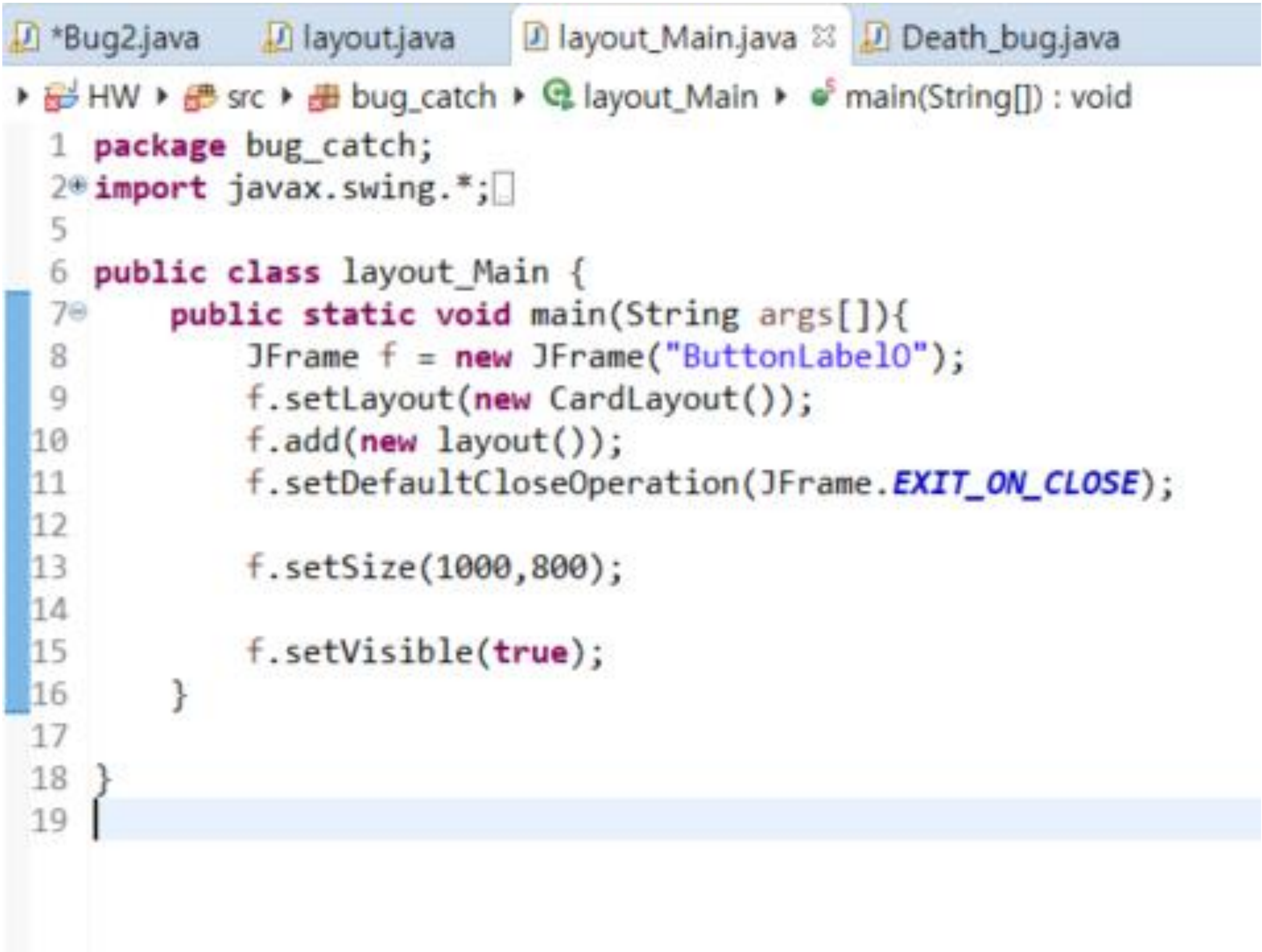
//성공 패널

구현 목표 - 레이아웃

4

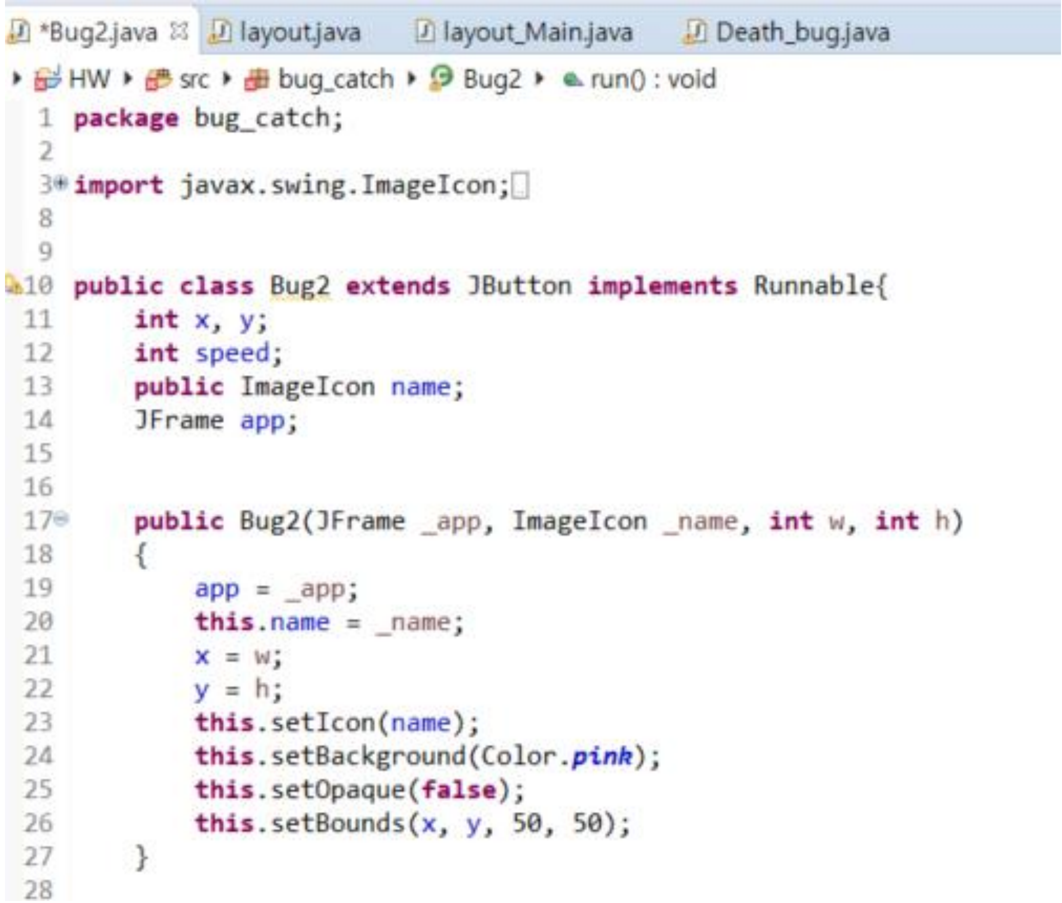
//실패 패널

구현 코드 - layout_Main

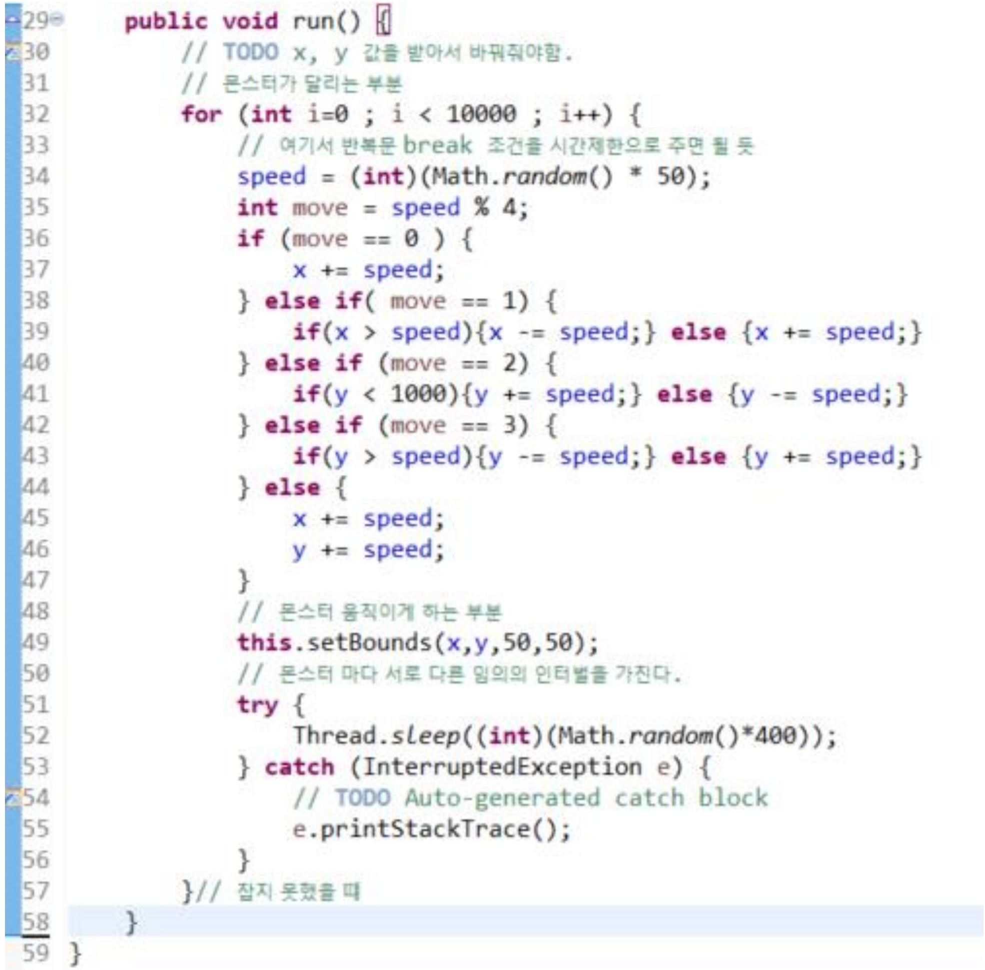


```java
package bug_catch;
import javax.swing.*;

public class layout_Main {
    public static void main(String args[]){
        JFrame f = new JFrame("ButtonLabel0");
        f.setLayout(new CardLayout());
        f.add(new layout());
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        f.setSize(1000,800);

        f.setVisible(true);
    }

}
```

구현 코드 Bug

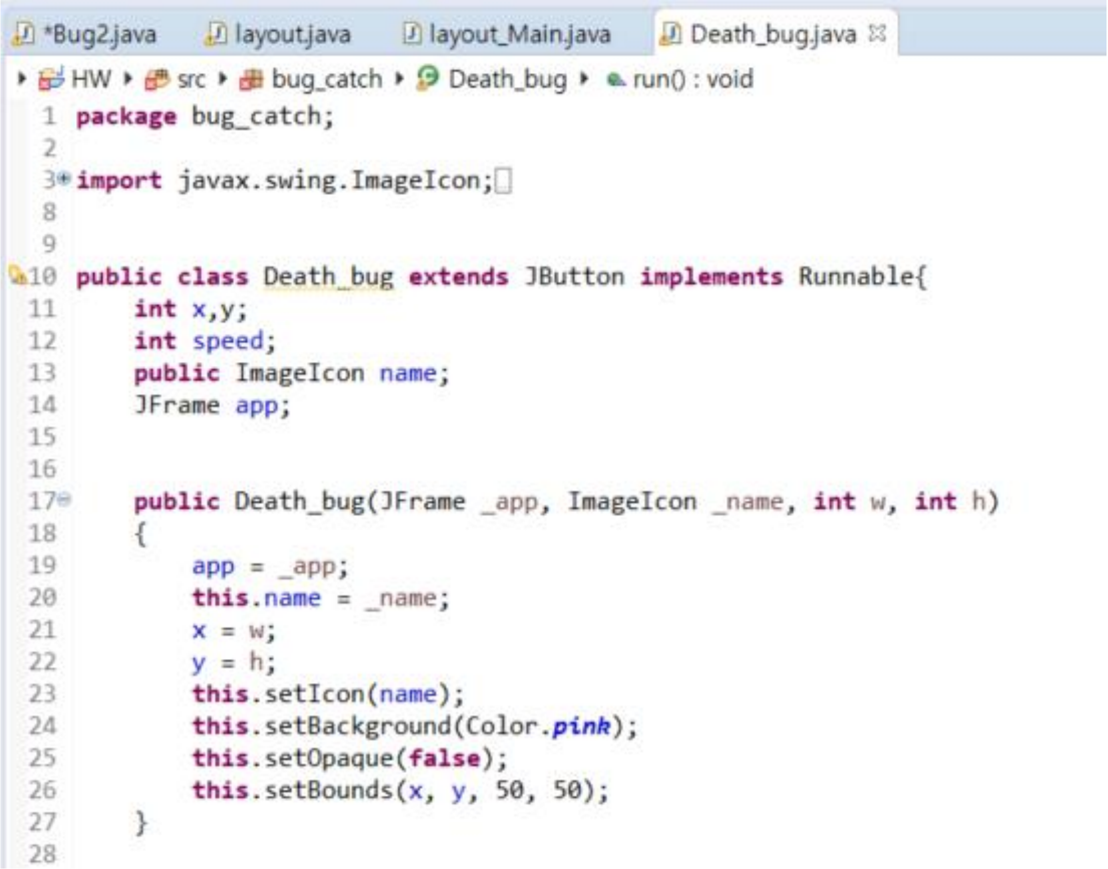


```java
package bug_catch;

import javax.swing.ImageIcon;


public class Bug2 extends JButton implements Runnable{
    int x, y;
    int speed;
    public ImageIcon name;
    JFrame app;


    public Bug2(JFrame _app, ImageIcon _name, int w, int h)
    {
        app = _app;
        this.name = _name;
        x = w;
        y = h;
        this.setIcon(name);
        this.setBackground(Color.pink);
        this.setOpaque(false);
        this.setBounds(x, y, 50, 50);
    }

```

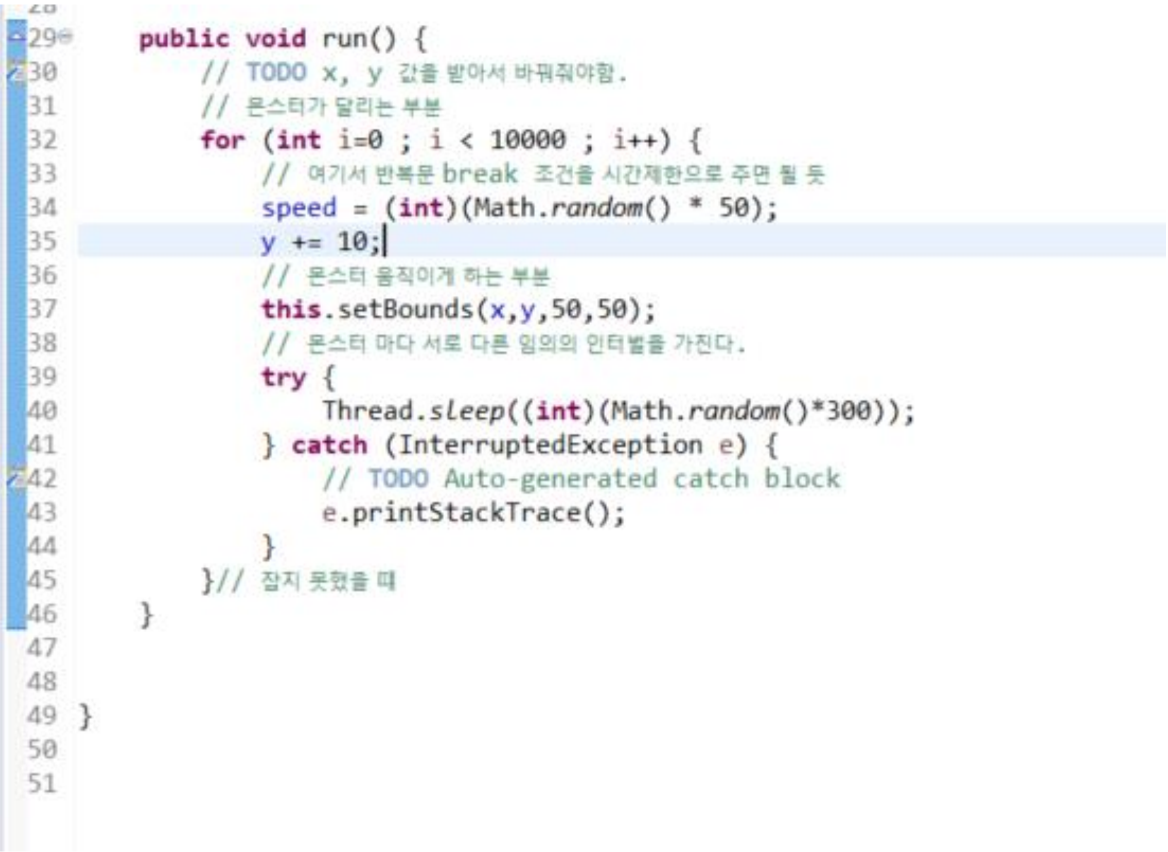


```java
    public void run() {
        // TODO x, y 값을 받아서 바꿔줘야함.
        // 몬스터가 달리는 부분
        for (int i=0 ; i < 10000 ; i++) {
            // 여기서 반복문 break 조건을 시간제한으로 주면 될 듯
            speed = (int)(Math.random() * 50);
            int move = speed % 4;
            if (move == 0 ) {
                x += speed;
            } else if( move == 1) {
                if(x > speed){x -= speed;} else {x += speed;}
            } else if (move == 2) {
                if(y < 1000){y += speed;} else {y -= speed;}
            } else if (move == 3) {
                if(y > speed){y -= speed;} else {y += speed;}
            } else {
                x += speed;
                y += speed;
            }
            // 몬스터 움직이게 하는 부분
            this.setBounds(x,y,50,50);
            // 몬스터 마다 서로 다른 임의의 인터벌을 가진다.
            try {
                Thread.sleep((int)(Math.random()*400));
            } catch (InterruptedException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }// 잡지 못했을 때
    }
}
```

구현 코드 - Death_bug

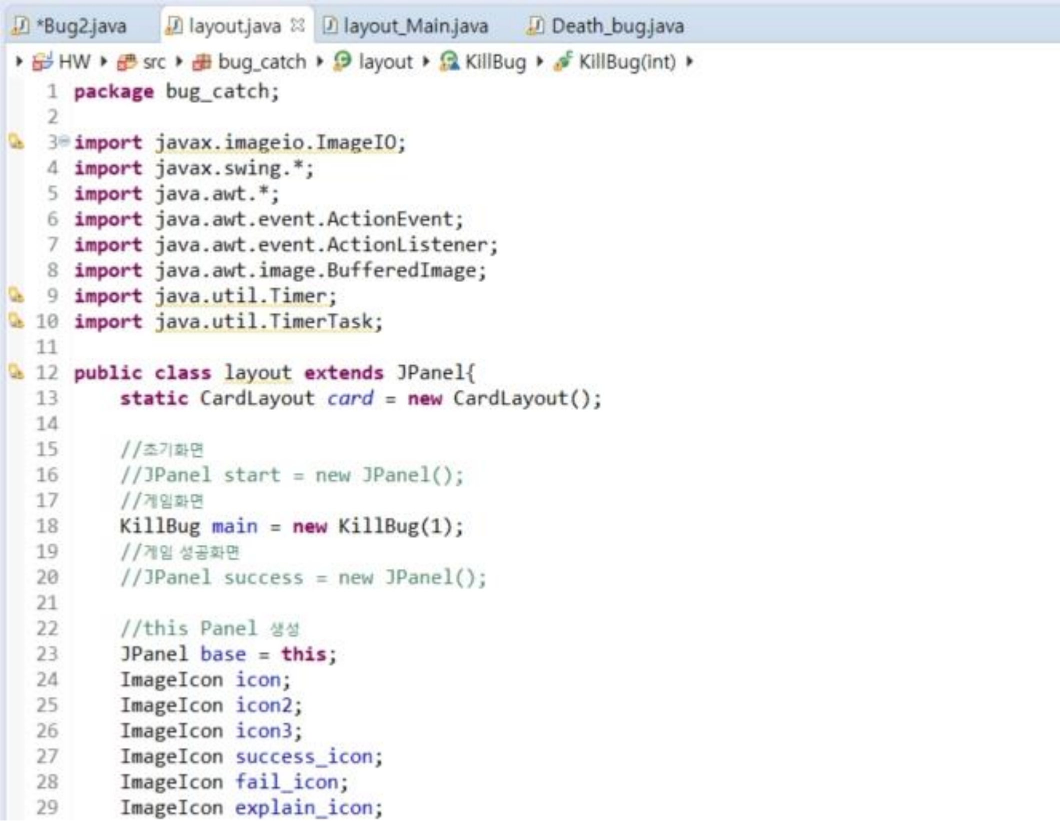


```java
package bug_catch;

import javax.swing.ImageIcon;


public class Death_bug extends JButton implements Runnable{
	int x,y;
	int speed;
	public ImageIcon name;
	JFrame app;


	public Death_bug(JFrame _app, ImageIcon _name, int w, int h)
	{
		app = _app;
		this.name = _name;
		x = w;
		y = h;
		this.setIcon(name);
		this.setBackground(Color.pink);
		this.setOpaque(false);
		this.setBounds(x, y, 50, 50);
	}

```

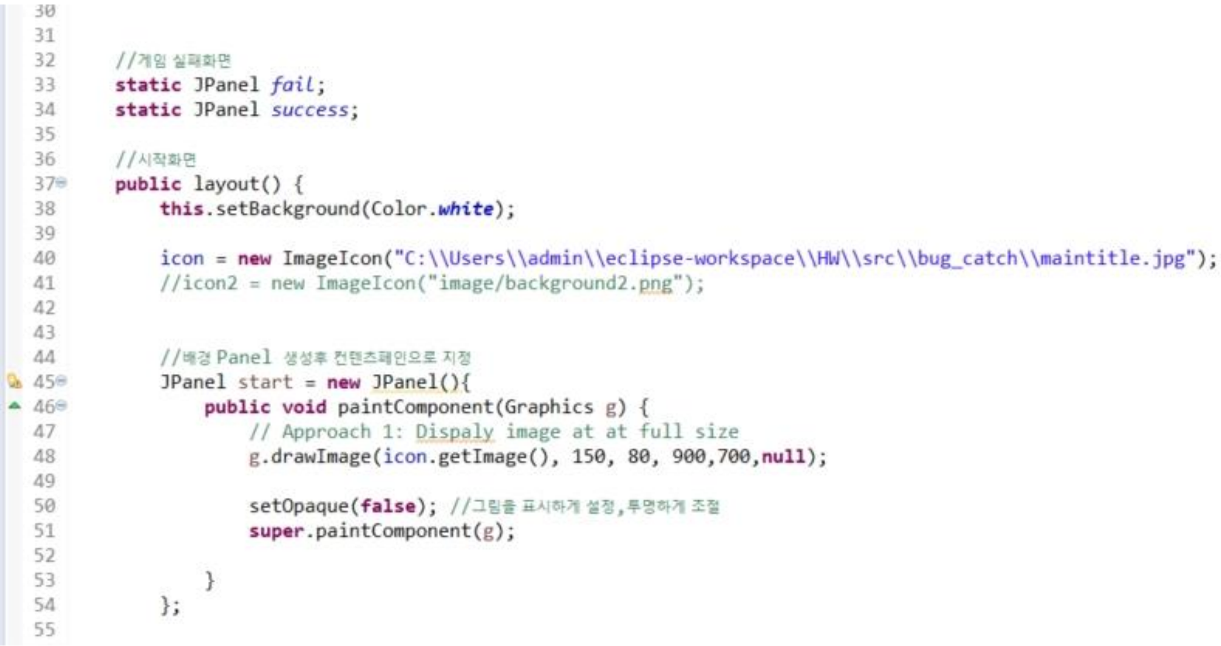


```java
	public void run() {
		// TODO x, y 값을 받아서 바꿔줘야함.
		// 몬스터가 달리는 부분
		for (int i=0 ; i < 10000 ; i++) {
			// 여기서 반복문 break 조건을 시간제한으로 주면 될 듯
			speed = (int)(Math.random() * 50);
			y += 10;
			// 몬스터 움직이게 하는 부분
			this.setBounds(x,y,50,50);
			// 몬스터 마다 서로 다른 임의의 인터벌을 가진다.
			try {
				Thread.sleep((int)(Math.random()*300));
			} catch (InterruptedException e) {
				// TODO Auto-generated catch block
				e.printStackTrace();
			}
		}// 잡지 못했을 때
	}


}


```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

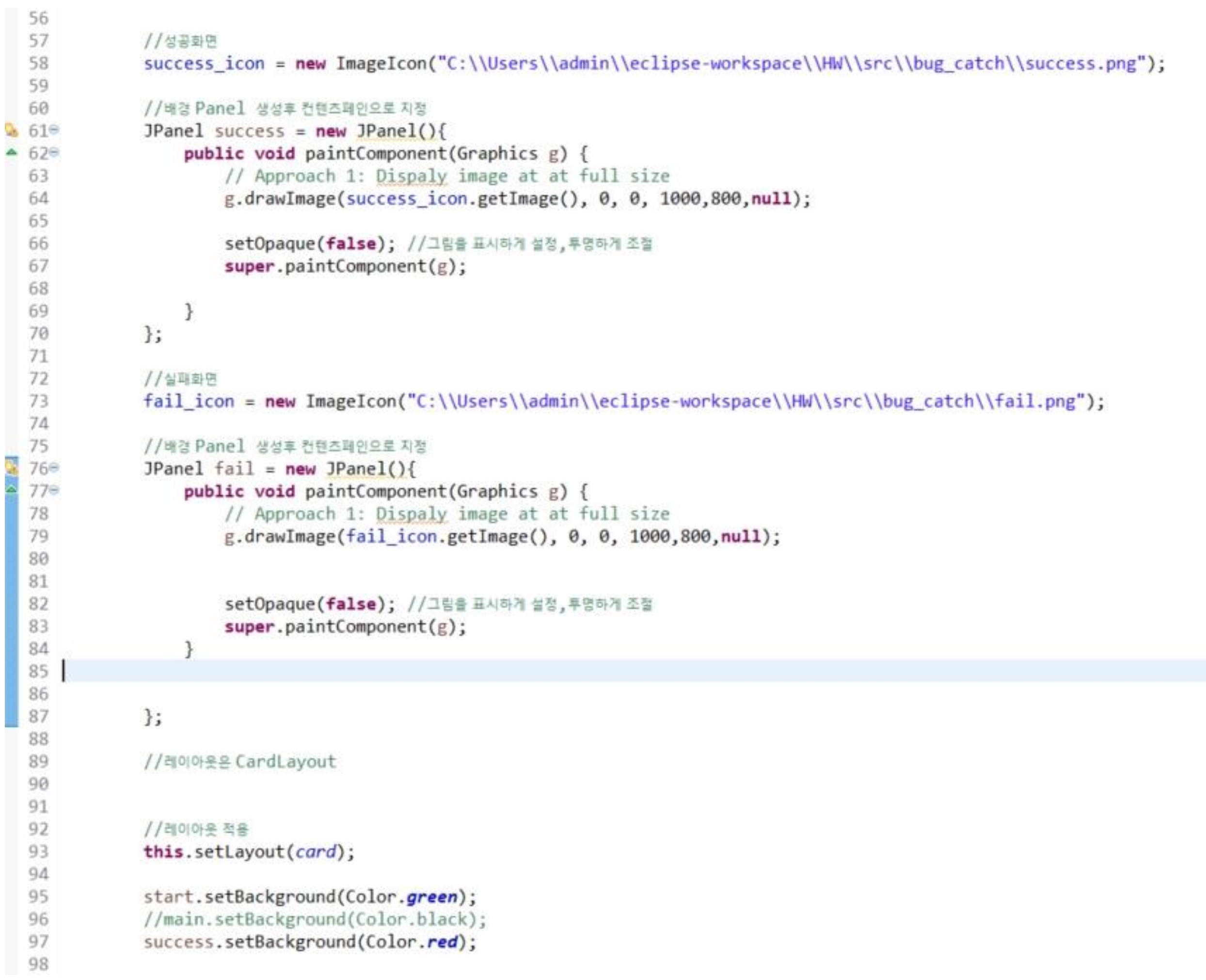


```java
package bug_catch;

import javax.imageio.ImageIO;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.image.BufferedImage;
import java.util.Timer;
import java.util.TimerTask;

public class layout extends JPanel{
    static CardLayout card = new CardLayout();

    //초기화면
    //JPanel start = new JPanel();
    //게임화면
    KillBug main = new KillBug(1);
    //게임 성공화면
    //JPanel success = new JPanel();

    //this Panel 생성
    JPanel base = this;
    ImageIcon icon;
    ImageIcon icon2;
    ImageIcon icon3;
    ImageIcon success_icon;
    ImageIcon fail_icon;
    ImageIcon explain_icon;
```

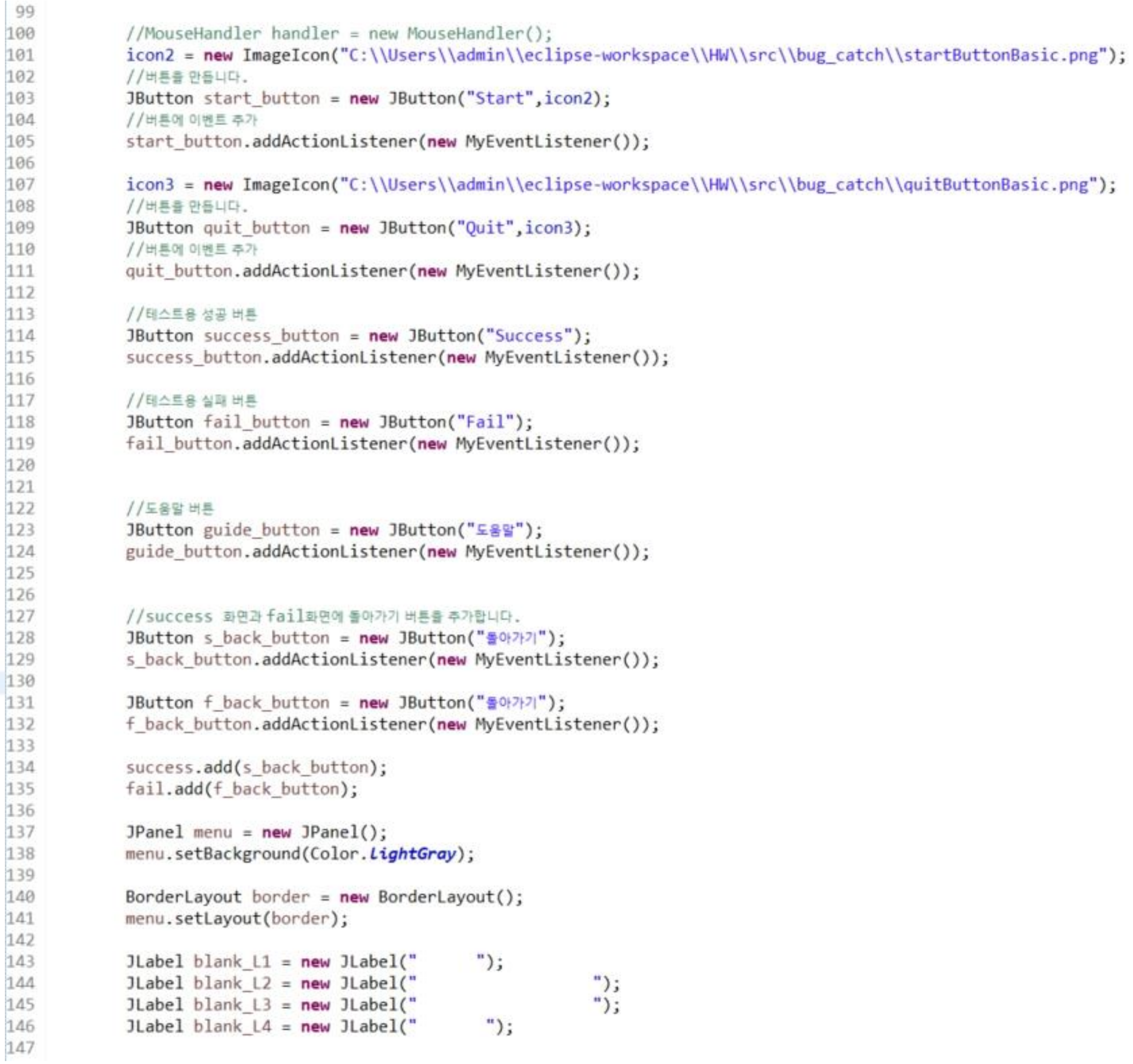


```java


    //게임 실패화면
    static JPanel fail;
    static JPanel success;

    //시작화면
    public layout() {
        this.setBackground(Color.white);

        icon = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\maintitle.jpg");
        //icon2 = new ImageIcon("image/background2.png");


        //배경 Panel 생성후 컨텐츠페인으로 지정
        JPanel start = new JPanel(){
            public void paintComponent(Graphics g) {
                // Approach 1: Dispaly image at at full size
                g.drawImage(icon.getImage(), 150, 80, 900,700,null);

                setOpaque(false); //그림을 표시하게 설정,투명하게 조절
                super.paintComponent(g);

            }
        };

```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

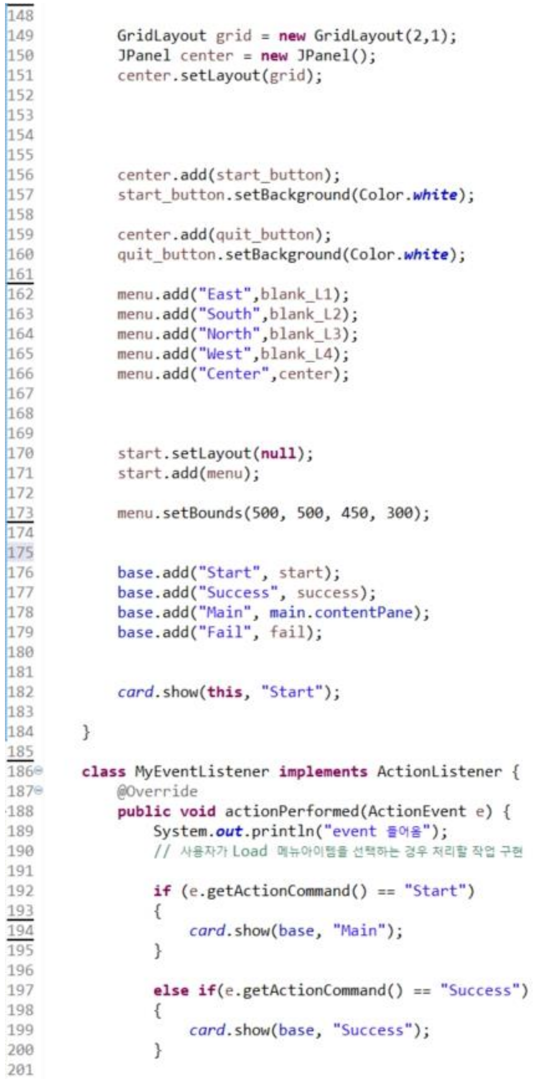


```java
        //성공화면
        success_icon = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\success.png");

        //배경 Panel 생성후 컨텐츠페인으로 지정
        JPanel success = new JPanel(){
            public void paintComponent(Graphics g) {
                // Approach 1: Dispaly image at at full size
                g.drawImage(success_icon.getImage(), 0, 0, 1000,800,null);

                setOpaque(false); //그림을 표시하게 설정,투명하게 조절
                super.paintComponent(g);

            }
        };

        //실패화면
        fail_icon = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\fail.png");

        //배경 Panel 생성후 컨텐츠페인으로 지정
        JPanel fail = new JPanel(){
            public void paintComponent(Graphics g) {
                // Approach 1: Dispaly image at at full size
                g.drawImage(fail_icon.getImage(), 0, 0, 1000,800,null);


                setOpaque(false); //그림을 표시하게 설정,투명하게 조절
                super.paintComponent(g);
            }


        };

        //레이아웃은 CardLayout


        //레이아웃 적용
        this.setLayout(card);

        start.setBackground(Color.green);
        //main.setBackground(Color.black);
        success.setBackground(Color.red);

```

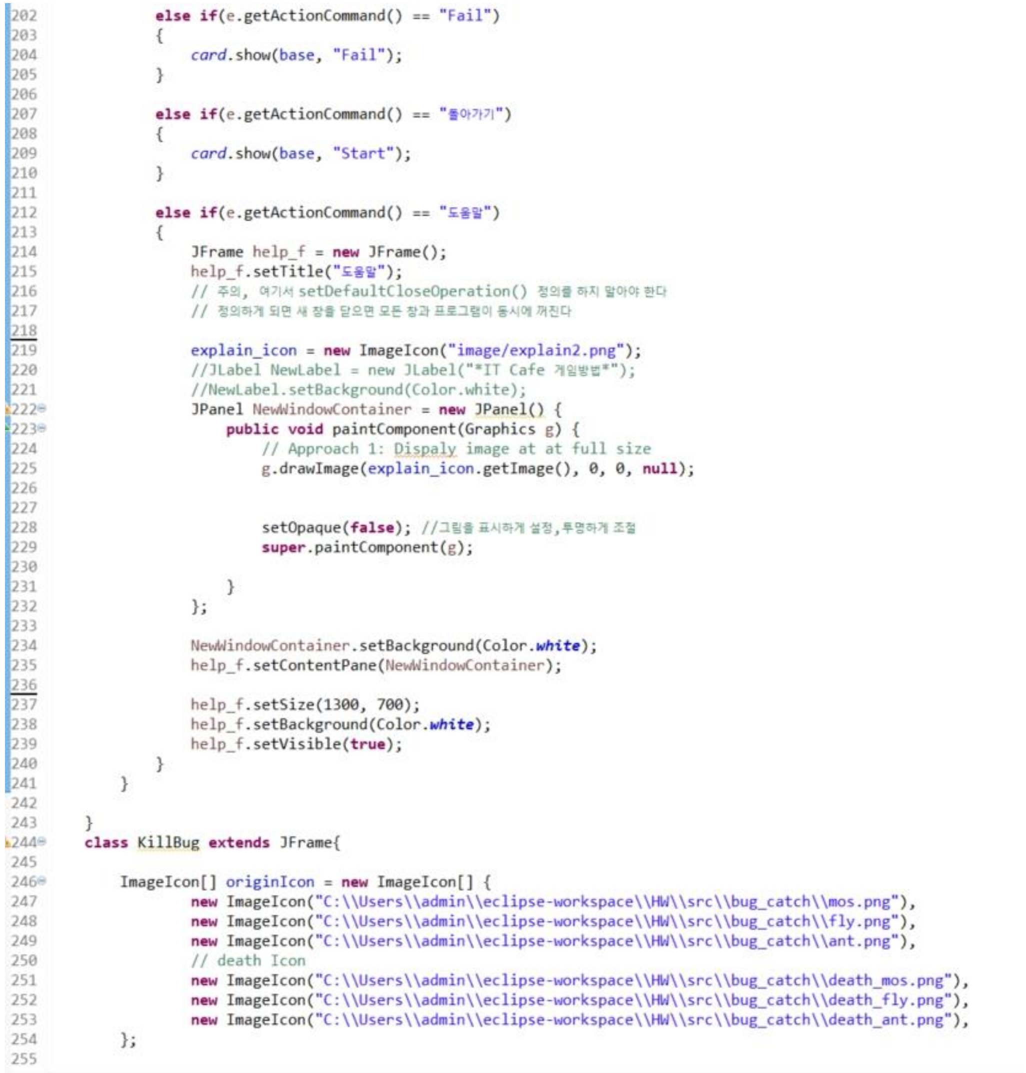


```java

        //MouseHandler handler = new MouseHandler();
        icon2 = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\startButtonBasic.png");
        //버튼을 만듭니다.
        JButton start_button = new JButton("Start",icon2);
        //버튼에 이벤트 추가
        start_button.addActionListener(new MyEventListener());

        icon3 = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\quitButtonBasic.png");
        //버튼을 만듭니다.
        JButton quit_button = new JButton("Quit",icon3);
        //버튼에 이벤트 추가
        quit_button.addActionListener(new MyEventListener());

        //테스트용 성공 버튼
        JButton success_button = new JButton("Success");
        success_button.addActionListener(new MyEventListener());

        //테스트용 실패 버튼
        JButton fail_button = new JButton("Fail");
        fail_button.addActionListener(new MyEventListener());


        //도움말 버튼
        JButton guide_button = new JButton("도움말");
        guide_button.addActionListener(new MyEventListener());


        //success 화면과 fail화면에 돌아가기 버튼을 추가합니다.
        JButton s_back_button = new JButton("돌아가기");
        s_back_button.addActionListener(new MyEventListener());

        JButton f_back_button = new JButton("돌아가기");
        f_back_button.addActionListener(new MyEventListener());

        success.add(s_back_button);
        fail.add(f_back_button);

        JPanel menu = new JPanel();
        menu.setBackground(Color.LightGray);

        BorderLayout border = new BorderLayout();
        menu.setLayout(border);

        JLabel blank_L1 = new JLabel("      ");
        JLabel blank_L2 = new JLabel("                  ");
        JLabel blank_L3 = new JLabel("                  ");
        JLabel blank_L4 = new JLabel("       ");

```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

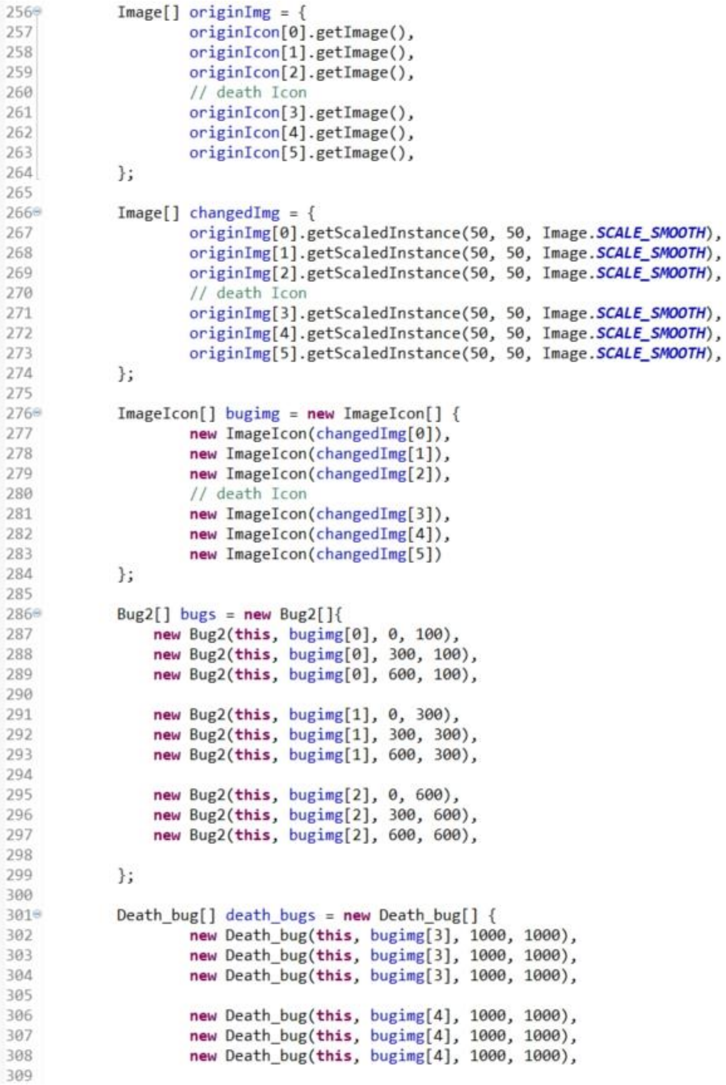


```java

        GridLayout grid = new GridLayout(2,1);
        JPanel center = new JPanel();
        center.setLayout(grid);




        center.add(start_button);
        start_button.setBackground(Color.white);

        center.add(quit_button);
        quit_button.setBackground(Color.white);

        menu.add("East",blank_L1);
        menu.add("South",blank_L2);
        menu.add("North",blank_L3);
        menu.add("West",blank_L4);
        menu.add("Center",center);



        start.setLayout(null);
        start.add(menu);

        menu.setBounds(500, 500, 450, 300);


        base.add("Start", start);
        base.add("Success", success);
        base.add("Main", main.contentPane);
        base.add("Fail", fail);


        card.show(this, "Start");

    }

    class MyEventListener implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent e) {
            System.out.println("event 들어옴");
            // 사용자가 Load 메뉴아이템을 선택하는 경우 처리할 작업 구현

            if (e.getActionCommand() == "Start")
            {
                card.show(base, "Main");
            }

            else if(e.getActionCommand() == "Success")
            {
                card.show(base, "Success");
            }

```

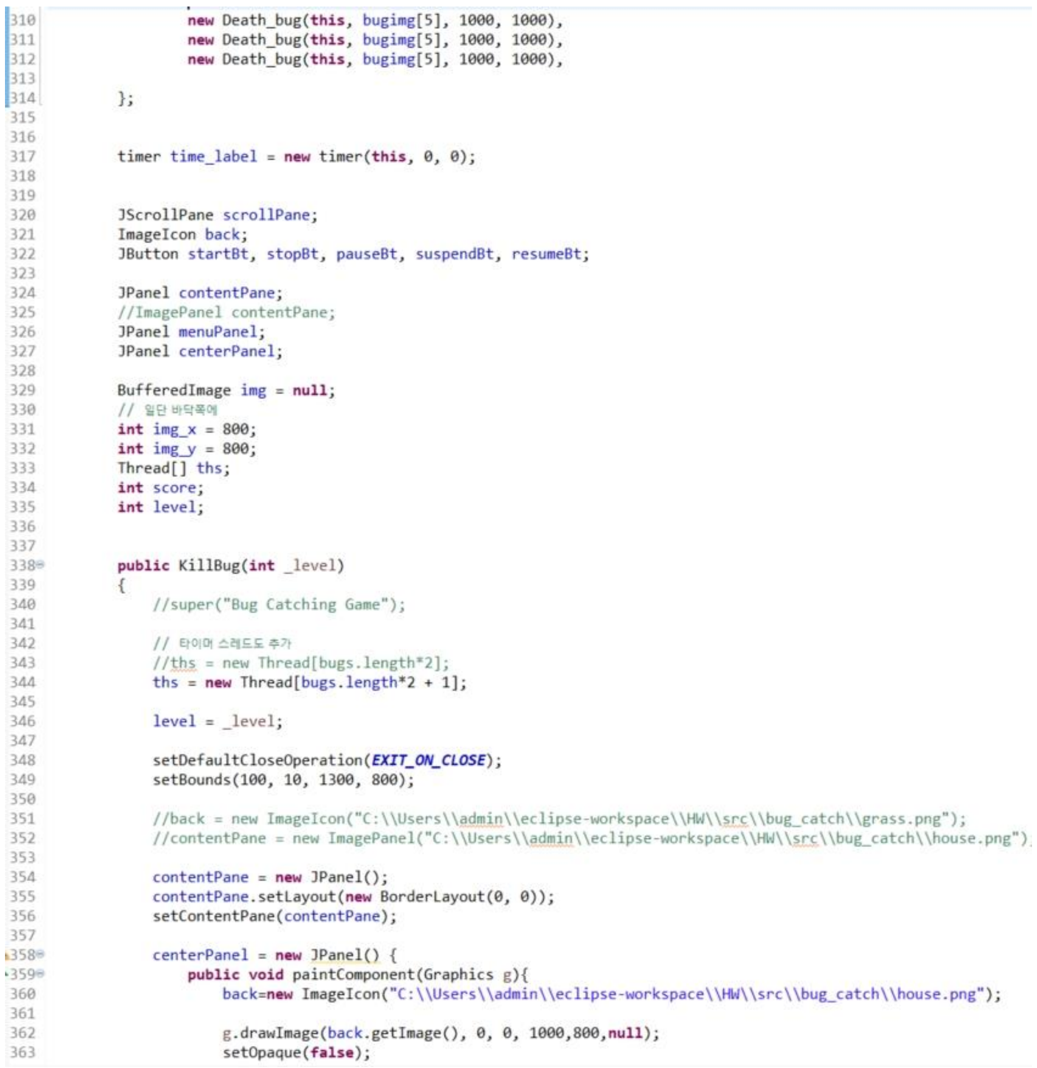


```java
            else if(e.getActionCommand() == "Fail")
            {
                card.show(base, "Fail");
            }

            else if(e.getActionCommand() == "돌아가기")
            {
                card.show(base, "Start");
            }

            else if(e.getActionCommand() == "도움말")
            {
                JFrame help_f = new JFrame();
                help_f.setTitle("도움말");
                // 주의, 여기서 setDefaultCloseOperation() 정의를 하지 말아야 한다
                // 정의하게 되면 새 창을 닫으면 모든 창과 프로그램이 동시에 꺼진다

                explain_icon = new ImageIcon("image/explain2.png");
                //JLabel NewLabel = new JLabel("*IT Cafe 게임방법*");
                //NewLabel.setBackground(Color.white);
                JPanel NewWindowContainer = new JPanel() {
                    public void paintComponent(Graphics g) {
                        // Approach 1: Dispaly image at at full size
                        g.drawImage(explain_icon.getImage(), 0, 0, null);


                        setOpaque(false); //그림을 표시하게 설정,투명하게 조절
                        super.paintComponent(g);

                    }
                };

                NewWindowContainer.setBackground(Color.white);
                help_f.setContentPane(NewWindowContainer);

                help_f.setSize(1300, 700);
                help_f.setBackground(Color.white);
                help_f.setVisible(true);
            }
        }

    }
    class KillBug extends JFrame{

        ImageIcon[] originIcon = new ImageIcon[] {
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\mos.png"),
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\fly.png"),
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\ant.png"),
                // death Icon
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\death_mos.png"),
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\death_fly.png"),
                new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\death_ant.png"),
        };

```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

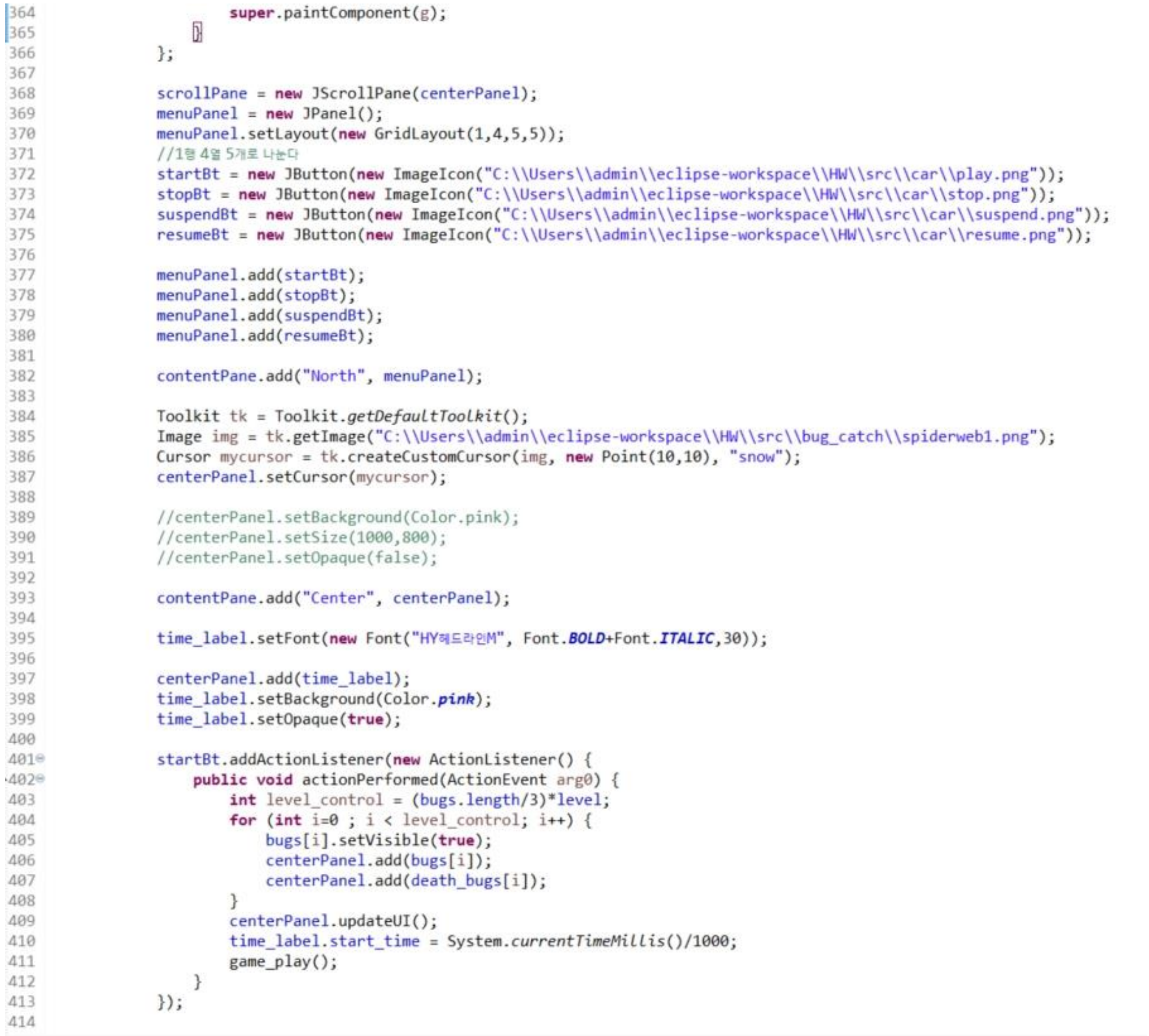


```java
Image[] originImg = {
    originIcon[0].getImage(),
    originIcon[1].getImage(),
    originIcon[2].getImage(),
    // death Icon
    originIcon[3].getImage(),
    originIcon[4].getImage(),
    originIcon[5].getImage(),
};

Image[] changedImg = {
    originImg[0].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
    originImg[1].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
    originImg[2].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
    // death Icon
    originImg[3].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
    originImg[4].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
    originImg[5].getScaledInstance(50, 50, Image.SCALE_SMOOTH),
};

ImageIcon[] bugimg = new ImageIcon[] {
    new ImageIcon(changedImg[0]),
    new ImageIcon(changedImg[1]),
    new ImageIcon(changedImg[2]),
    // death Icon
    new ImageIcon(changedImg[3]),
    new ImageIcon(changedImg[4]),
    new ImageIcon(changedImg[5])
};

Bug2[] bugs = new Bug2[]{
  new Bug2(this, bugimg[0], 0, 100),
  new Bug2(this, bugimg[0], 300, 100),
  new Bug2(this, bugimg[0], 600, 100),

  new Bug2(this, bugimg[1], 0, 300),
  new Bug2(this, bugimg[1], 300, 300),
  new Bug2(this, bugimg[1], 600, 300),

  new Bug2(this, bugimg[2], 0, 600),
  new Bug2(this, bugimg[2], 300, 600),
  new Bug2(this, bugimg[2], 600, 600),

};

Death_bug[] death_bugs = new Death_bug[] {
    new Death_bug(this, bugimg[3], 1000, 1000),
    new Death_bug(this, bugimg[3], 1000, 1000),
    new Death_bug(this, bugimg[3], 1000, 1000),

    new Death_bug(this, bugimg[4], 1000, 1000),
    new Death_bug(this, bugimg[4], 1000, 1000),
    new Death_bug(this, bugimg[4], 1000, 1000),

```

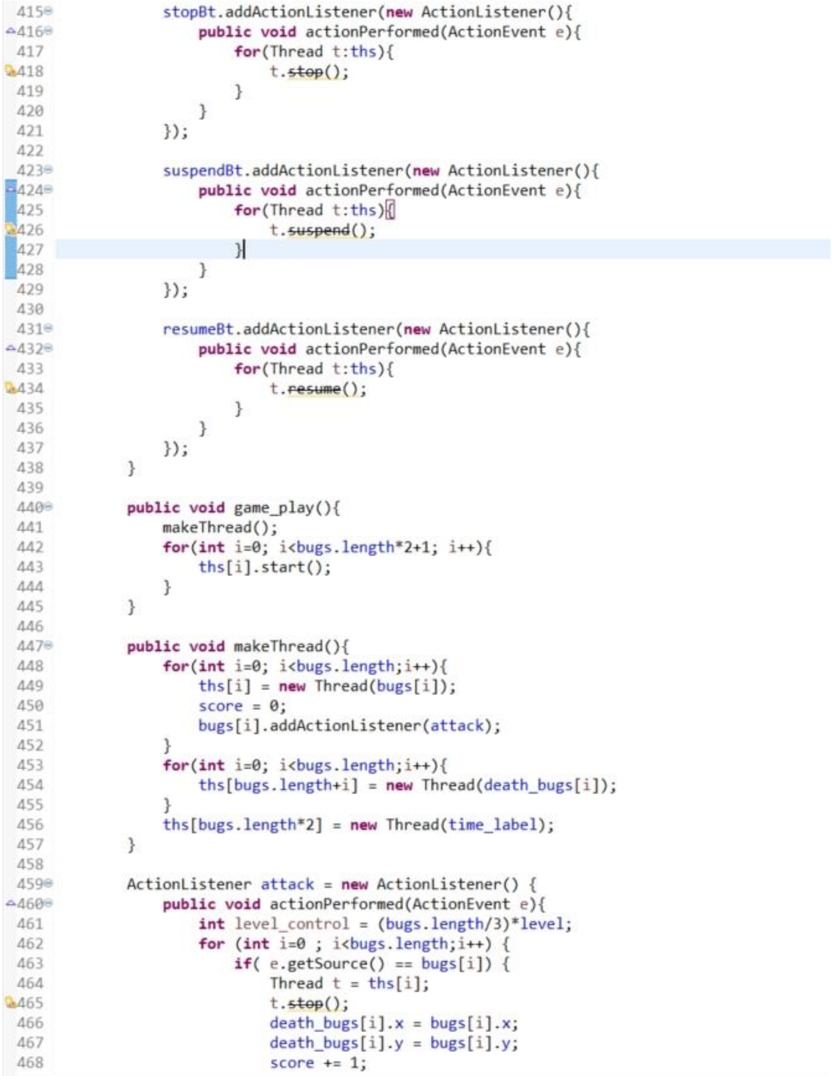


```java
    new Death_bug(this, bugimg[5], 1000, 1000),
    new Death_bug(this, bugimg[5], 1000, 1000),
    new Death_bug(this, bugimg[5], 1000, 1000),

};


timer time_label = new timer(this, 0, 0);


JScrollPane scrollPane;
ImageIcon back;
JButton startBt, stopBt, pauseBt, suspendBt, resumeBt;

JPanel contentPane;
//ImagePanel contentPane;
JPanel menuPanel;
JPanel centerPanel;

BufferedImage img = null;
// 일단 바닥쪽에
int img_x = 800;
int img_y = 800;
Thread[] ths;
int score;
int level;


public KillBug(int _level)
{
  //super("Bug Catching Game");

  // 타이머 스레드도 추가
  //ths = new Thread[bugs.length*2];
  ths = new Thread[bugs.length*2 + 1];

  level = _level;

  setDefaultCloseOperation(EXIT_ON_CLOSE);
  setBounds(100, 10, 1300, 800);

  //back = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\grass.png");
  //contentPane = new ImagePanel("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\house.png")

  contentPane = new JPanel();
  contentPane.setLayout(new BorderLayout(0, 0));
  setContentPane(contentPane);

  centerPanel = new JPanel() {
    public void paintComponent(Graphics g){
      back=new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\house.png");

      g.drawImage(back.getImage(), 0, 0, 1000,800,null);
      setOpaque(false);
```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

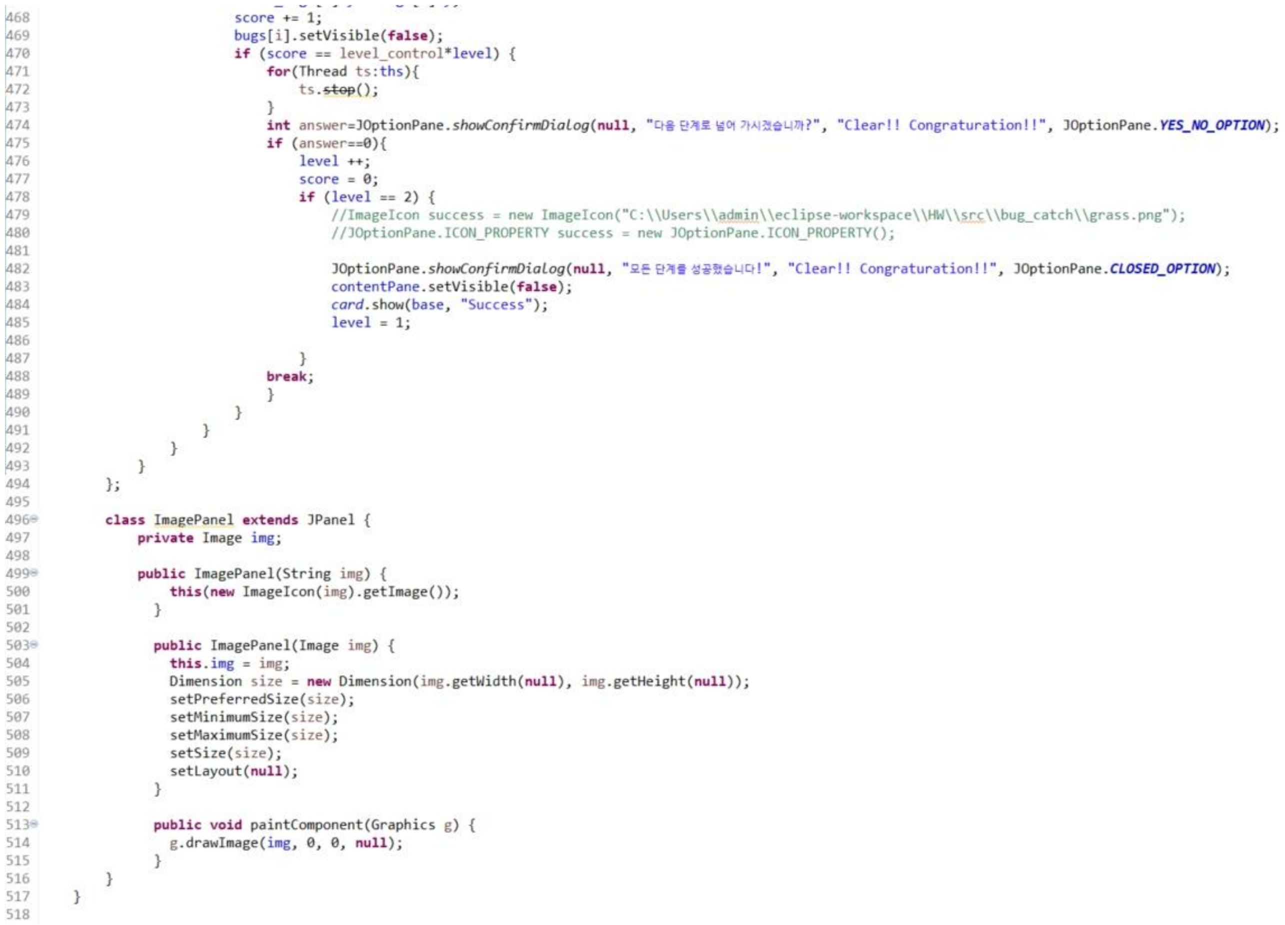


```java
                super.paintComponent(g);
            }
        };

        scrollPane = new JScrollPane(centerPanel);
        menuPanel = new JPanel();
        menuPanel.setLayout(new GridLayout(1,4,5,5));
        //1행 4열 5개로 나눈다
        startBt = new JButton(new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\car\\play.png"));
        stopBt = new JButton(new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\car\\stop.png"));
        suspendBt = new JButton(new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\car\\suspend.png"));
        resumeBt = new JButton(new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\car\\resume.png"));

        menuPanel.add(startBt);
        menuPanel.add(stopBt);
        menuPanel.add(suspendBt);
        menuPanel.add(resumeBt);

        contentPane.add("North", menuPanel);

        Toolkit tk = Toolkit.getDefaultToolkit();
        Image img = tk.getImage("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\spiderweb1.png");
        Cursor mycursor = tk.createCustomCursor(img, new Point(10,10), "snow");
        centerPanel.setCursor(mycursor);

        //centerPanel.setBackground(Color.pink);
        //centerPanel.setSize(1000,800);
        //centerPanel.setOpaque(false);

        contentPane.add("Center", centerPanel);

        time_label.setFont(new Font("HY헤드라인M", Font.BOLD+Font.ITALIC,30));

        centerPanel.add(time_label);
        time_label.setBackground(Color.pink);
        time_label.setOpaque(true);

        startBt.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                int level_control = (bugs.length/3)*level;
                for (int i=0 ; i < level_control; i++) {
                    bugs[i].setVisible(true);
                    centerPanel.add(bugs[i]);
                    centerPanel.add(death_bugs[i]);
                }
                centerPanel.updateUI();
                time_label.start_time = System.currentTimeMillis()/1000;
                game_play();
            }
        });
```

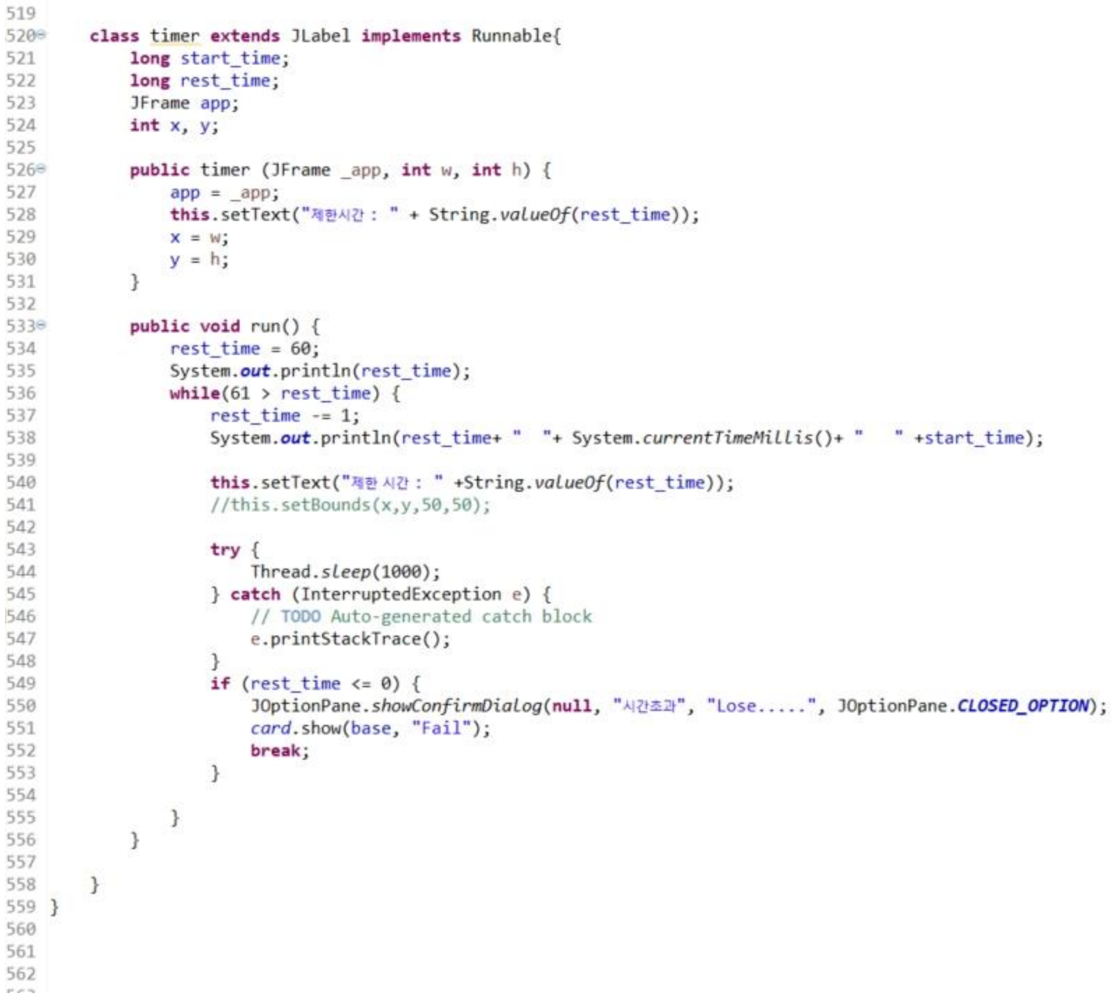


```java
        stopBt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                for(Thread t:ths){
                    t.stop();
                }
            }
        });

        suspendBt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                for(Thread t:ths){
                    t.suspend();
                }
            }
        });

        resumeBt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                for(Thread t:ths){
                    t.resume();
                }
            }
        });
    }

    public void game_play(){
        makeThread();
        for(int i=0; i<bugs.length*2+1; i++){
            ths[i].start();
        }
    }

    public void makeThread(){
        for(int i=0; i<bugs.length;i++){
            ths[i] = new Thread(bugs[i]);
            score = 0;
            bugs[i].addActionListener(attack);
        }
        for(int i=0; i<bugs.length;i++){
            ths[bugs.length+i] = new Thread(death_bugs[i]);
        }
        ths[bugs.length*2] = new Thread(time_label);
    }

    ActionListener attack = new ActionListener() {
        public void actionPerformed(ActionEvent e){
            int level_control = (bugs.length/3)*level;
            for (int i=0 ; i<bugs.length;i++) {
                if( e.getSource() == bugs[i]) {
                    Thread t = ths[i];
                    t.stop();
                    death_bugs[i].x = bugs[i].x;
                    death_bugs[i].y = bugs[i].y;
                    score += 1;
```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)

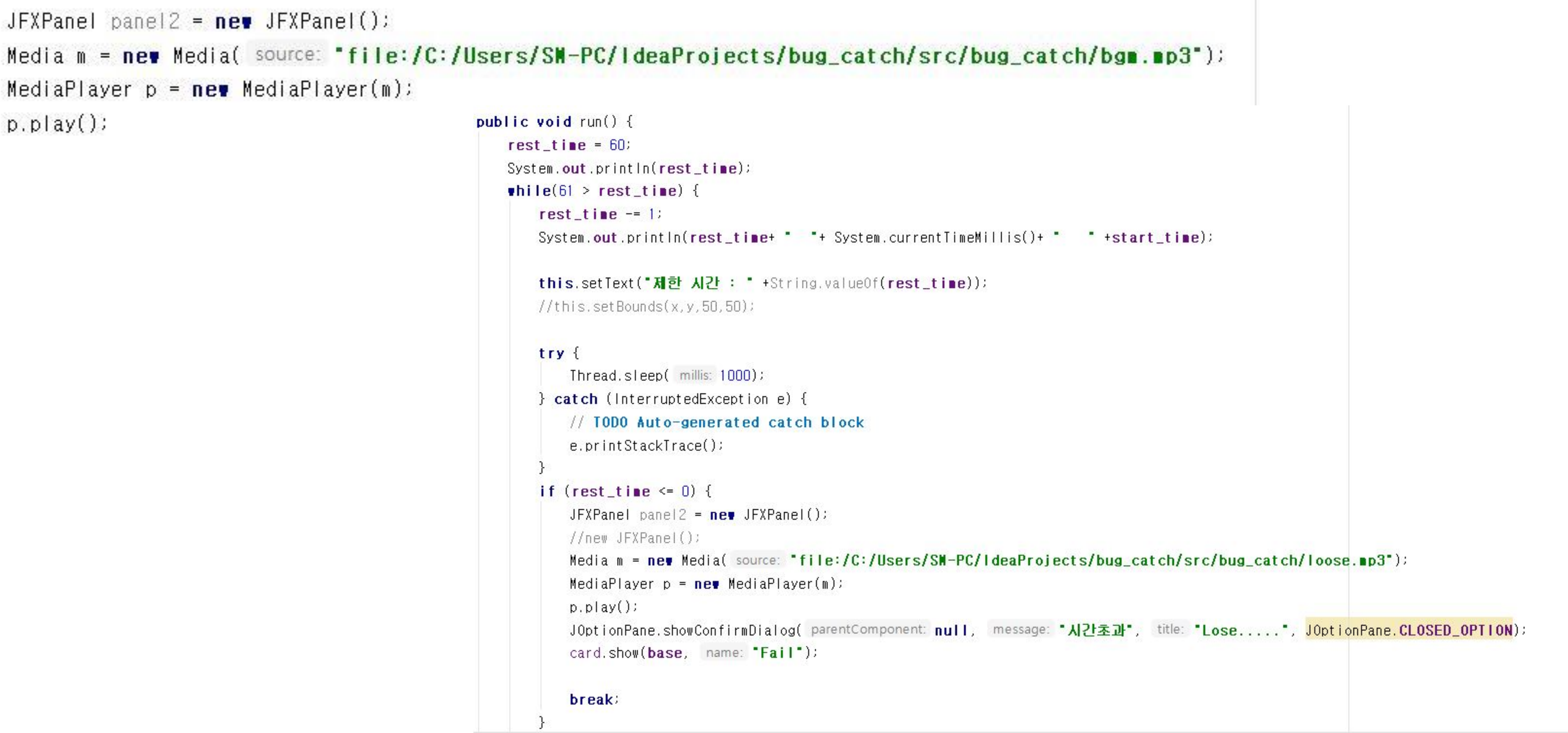


```
                    score += 1;
                    bugs[i].setVisible(false);
                    if (score == level_control*level) {
                        for(Thread ts:ths){
                            ts.~~stop~~();
                        }
                        int answer=JOptionPane.showConfirmDialog(null, "다음 단계로 넘어 가시겠습니까?", "Clear!! Congraturation!!", JOptionPane.YES_NO_OPTION);
                        if (answer==0){
                            level ++;
                            score = 0;
                            if (level == 2) {
                                //ImageIcon success = new ImageIcon("C:\\Users\\admin\\eclipse-workspace\\HW\\src\\bug_catch\\grass.png");
                                //JOptionPane.ICON_PROPERTY success = new JOptionPane.ICON_PROPERTY();

                                JOptionPane.showConfirmDialog(null, "모든 단계를 성공했습니다!", "Clear!! Congraturation!!", JOptionPane.CLOSED_OPTION);
                                contentPane.setVisible(false);
                                card.show(base, "Success");
                                level = 1;

                            }
                        break;
                        }
                    }
                }
            }
        }
    };

    class ImagePanel extends JPanel {
        private Image img;

        public ImagePanel(String img) {
            this(new ImageIcon(img).getImage());
          }

          public ImagePanel(Image img) {
            this.img = img;
            Dimension size = new Dimension(img.getWidth(null), img.getHeight(null));
            setPreferredSize(size);
            setMinimumSize(size);
            setMaximumSize(size);
            setSize(size);
            setLayout(null);
          }

          public void paintComponent(Graphics g) {
            g.drawImage(img, 0, 0, null);
          }
    }
}
```



```
    class timer extends JLabel implements Runnable{
        long start_time;
        long rest_time;
        JFrame app;
        int x, y;

        public timer (JFrame _app, int w, int h) {
            app = _app;
            this.setText("제한시간 : " + String.valueOf(rest_time));
            x = w;
            y = h;
        }

        public void run() {
            rest_time = 60;
            System.out.println(rest_time);
            while(61 > rest_time) {
                rest_time -= 1;
                System.out.println(rest_time+ "  "+ System.currentTimeMillis()+ "   " +start_time);

                this.setText("제한 시간 : " +String.valueOf(rest_time));
                //this.setBounds(x,y,50,50);

                try {
                    Thread.sleep(1000);
                } catch (InterruptedException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                if (rest_time <= 0) {
                    JOptionPane.showConfirmDialog(null, "시간초과", "Lose.....", JOptionPane.CLOSED_OPTION);
                    card.show(base, "Fail");
                    break;
                }

            }
        }

    }
}
```

구현 코드 - layout (전체 레이아웃 & KillBug & timer)



```java
JFXPanel panel2 = new JFXPanel();
Media m = new Media( source: "file:/C:/Users/SM-PC/IdeaProjects/bug_catch/src/bug_catch/bgm.mp3");
MediaPlayer p = new MediaPlayer(m);
p.play();
```

```java
public void run() {
    rest_time = 60;
    System.out.println(rest_time);
    while(61 > rest_time) {
        rest_time -= 1;
        System.out.println(rest_time+ "   "+ System.currentTimeMillis()+ "     " +start_time);

        this.setText("제한 시간 : " +String.valueOf(rest_time));
        //this.setBounds(x,y,50,50);

        try {
            Thread.sleep( millis: 1000);
        } catch (InterruptedException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        if (rest_time <= 0) {
            JFXPanel panel2 = new JFXPanel();
            //new JFXPanel();
            Media m = new Media( source: "file:/C:/Users/SM-PC/IdeaProjects/bug_catch/src/bug_catch/loose.mp3");
            MediaPlayer p = new MediaPlayer(m);
            p.play();
            JOptionPane.showConfirmDialog( parentComponent: null,  message: "시간초과",  title: "Lose.....", JOptionPane.CLOSED_OPTION);
            card.show(base,  name: "Fail");

            break;
        }
```

구현결과

시연

감사합니다