

객체지향프로그래밍 – 13주차 실습 활동지

성명: 장효택

학과: 미디어학과

학번: 201721070

반/그룹: B반

※ 본 실습활동지를 작성함에 있어 다른 학생의 문서로부터 일부 또는 전체를 복사하였습니까?

예() 아니오(v) (복사 하였다면 예에 체크하고 아니라면 아니오에 체크하시오)

1. ch10-part2의 ButtonTest (첨부파일) 를 이용하여 다음 물음에 답하시오.

가) ButtonFrame.java에서는 ColorAction inner class로 ActionListener를 구현하여 button의 listener로 등록한다. 이를 Lambda expression으로 수정하시오. (코드 및 실행화면 포함) [3점]

```
public ButtonFrame() {
    setSize(DEFAULT_WIDTH, DEFAULT_HEIGHT);

    JButton yellowButton = new JButton("Yellow");
    JButton blueButton = new JButton("Blue");
    JButton redButton = new JButton("Red");
    buttonPanel = new JPanel();

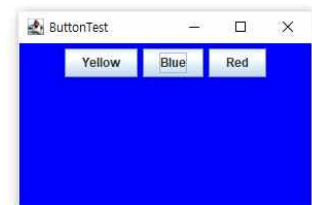
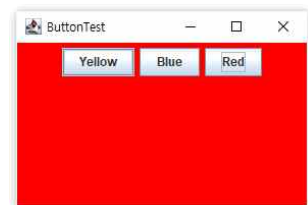
    buttonPanel.add(yellowButton);
    buttonPanel.add(blueButton);
    buttonPanel.add(redButton);

    add(buttonPanel);

    yellowButton.addActionListener(event -> {
        Color backgroundColor = Color.YELLOW;
        buttonPanel.setBackground(backgroundColor);
    });

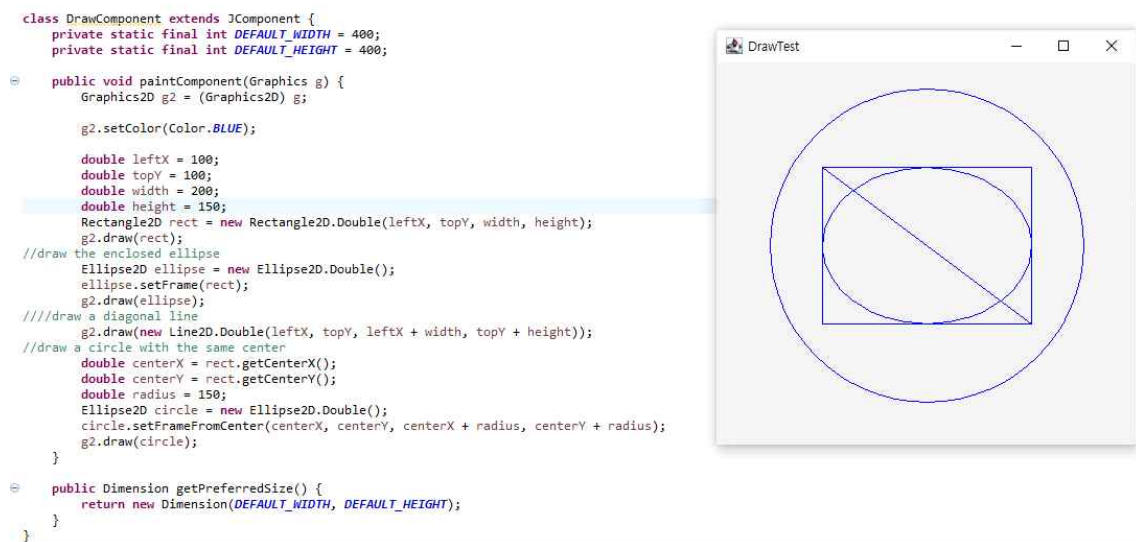
    blueButton.addActionListener(event -> {
        Color backgroundColor = Color.BLUE;
        buttonPanel.setBackground(backgroundColor);
    });

    redButton.addActionListener(event -> {
        Color backgroundColor = Color.RED;
        buttonPanel.setBackground(backgroundColor);
    });
}
```



2. ch10-part1의 DrawTest와 ch10-part3의 ActionTest (첨부파일)를 각각 수행해 본 후 다음 물음에 답하시오.

가) 모든 도형이 파란 색으로 그리도록 DrawTest.java를 수정하시오. (코드 포함 및 실행화면) [3점]



나) ActionFrame.java를 이용하여 다음과 같이 수정하시오. (코드 및 실행화면) [4점]

- ActionFrame.java에서 정의한 Frame에 button panel 외에 DrawTest.java에서 정의한 DrawComponent를 하나 만들어 추가한다.

Hint) button panel과 DrawComponent를 BorderLayout에서 NORTH와 CENTER에 배치한다.

- Button을 누를 때 마다 DrawComponent에 그려진 도형이 button의 color로 변경되도록 한다.

Hint)

- DrawComponent class를 ActionFrame class의 inner class로 만든다.
- button 클릭 event를 처리할 때 Color를 설정한 후 repaint가 되도록 한다

```

// panel to frame
setLayout(new BorderLayout());
add(buttonPanel, BorderLayout.NORTH);
DrawPanel.add(drawComponent);
add(DrawPanel, BorderLayout.CENTER);
pack();

// create the V, R, and D keys with names
public void actionPerformed(ActionEvent event) {

    Color c = (Color) getValue("color");
    drawComponent.setColor(c);
    repaint();
}

// end of inner class ColorAction

```

```

class DrawComponent extends JComponent {
    private static final int DEFAULT_WIDTH = 400;
    private static final int DEFAULT_HEIGHT = 400;
    Color c;
    public DrawComponent(Color c) {
        this.c = c;
    }

    public void setColor(Color c2) {
        this.c = c2;
    }

    public DrawComponent() {
        this.c = Color.BLUE;
    }

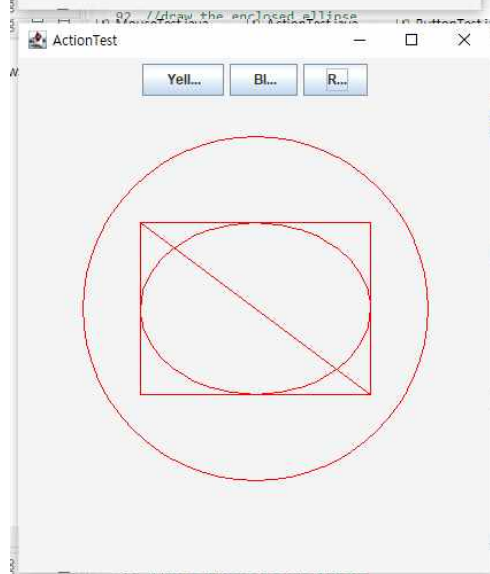
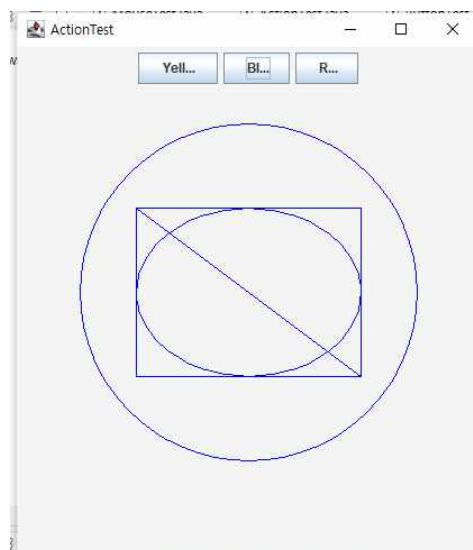
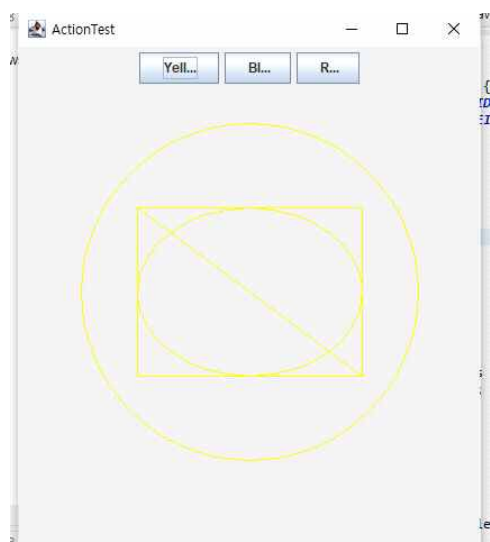
    public void paintComponent(Graphics g) {
        Graphics2D g2 = (Graphics2D) g;

        g2.setColor(c);

        double leftX = 100;
        double topY = 100;
        double width = 200;
        double height = 150;
        Rectangle2D rect = new Rectangle2D.Double(leftX, topY, width, height);
        g2.draw(rect);
        //draw the enclosed ellipse
        Ellipse2D ellipse = new Ellipse2D.Double();
        ellipse setFrame(rect);
        g2.draw(ellipse);
        //draw a diagonal line
        g2.draw(new Line2D.Double(leftX, topY, leftX + width, topY + height));
        //draw a circle with the same center
        double centerX = rect.getCenterX();
        double centerY = rect.getCenterY();
        double radius = 150;
        Ellipse2D circle = new Ellipse2D.Double();
        circle.setFrameFromCenter(centerX, centerY, centerX + radius, centerY + radius);
        g2.draw(circle);
    }

    public Dimension getPreferredSize() {
        return new Dimension(DEFAULT_WIDTH, DEFAULT_HEIGHT);
    }
}

```



3. ch10-part3의 강의노트 MouseTest (첨부파일)를 실행해 본 후 다음 물음에 답하시오.

가) double click할 때 사각형이 지워지는 대신 오른쪽 마우스 버튼을 누를 경우 지워지도록 수정하시오. (수정부분 코드 영역 포함) [2점]

```
public void mouseClicked(MouseEvent event) {  
    current = find(event.getPoint());  
    if (current != null)  
        if(event.getButton()==MouseEvent.BUTTON3)  
            remove(current);  
}
```



나) click하면 모든 사각형이 지워지는 Button을 추가하시오. (추가 또는 수정부분 코드 영역 포함) [2점]

Hint) 추가하는 Button은 BorderLayout에서 SOUTH에 위치한다.

```
public MouseComponent() {  
    setLayout(new BorderLayout());  
    squares = new ArrayList<>();  
    current = null;  
    addMouseListener(new MouseHandler());  
    addMouseMotionListener(new MouseMotionHandler());  
  
    removeButton = new JButton("REMOVE ALL");  
  
    JPanel jpanel = new JPanel();  
  
    jpanel.add(removeButton);  
  
    removeButton.addActionListener(event->{  
        squares.clear();  
        repaint();  
    });  
  
    add(jpanel, BorderLayout.SOUTH);  
}
```