

2017학년도 1학기
JAVA 프로그래밍 II

514770-1
2017년 봄학기
5/2/2017
박경신

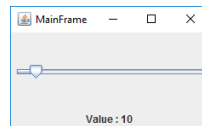
Lab #5 (GUI)

- 기존 요구사항 분석
 - Lab #4는 List<Photo>를 가지고 interface, collection과 lambda를 이용한 다양한 기능을 사용
 - Lab #5는 AWT/SWING을 이용한 GUI 프로그래밍
- Label, Slider, Textfield, Button, Panel, Layout
- Custom Panel, Custom ToggleButton

Lab #5_1 Slider (GUI)

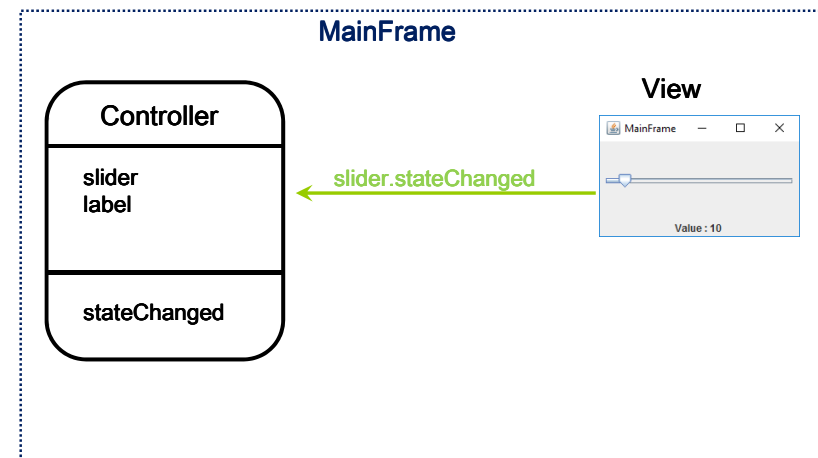
- Lab#5_1에서는 JLabel, JSlider를 이용한 슬라이더 값을 레이블에 출력하는 간단한 GUI를 구현한다.

```
public class MainFrame extends JFrame {  
    JSlider slider = new JSlider(JSlider.HORIZONTAL, 0, 100, 10);  
    JLabel label = new JLabel("Value : ", JLabel.CENTER);  
  
    slider.addChangeListener(new ChangeListener() {  
        public void stateChanged(ChangeEvent e) {  
            label.setText("Value : " + ((JSlider)e.getSource()).getValue());  
        }  
    });  
}
```



- Java2-lab5_1 폴더에 저장 후 제출

Lab #5_1 Slider (GUI)

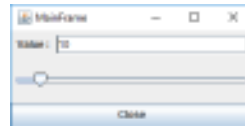


Lab #5_1 Slider (GUI)

□ Lab#5_1를 확장하여 JLabel, JSlider, JTextField, JButton를 이용한 슬라이더 값을 텍스트필드에 출력하는, 버튼으로 종료하는 간단한 GUI를 구현한다.

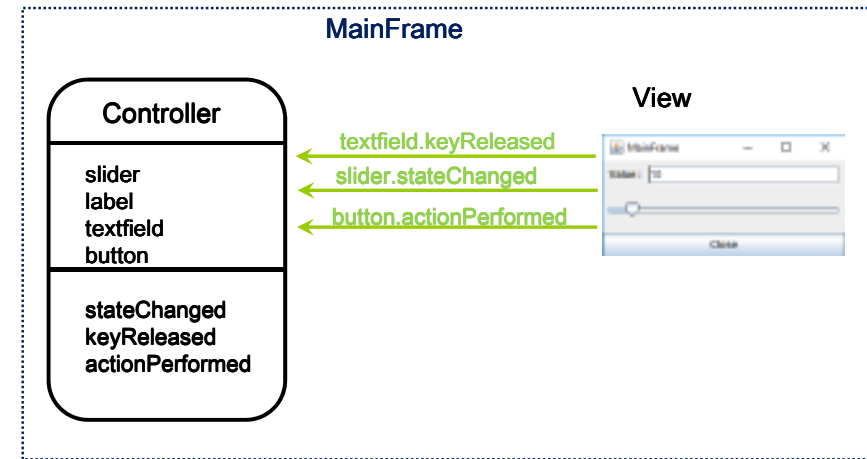
```

public class MainFrame extends JFrame implements ChangeListener, KeyListener,
ActionListener {
    JSlider slider = new JSlider(JSlider.HORIZONTAL, 0, 100, 10);
    JLabel label = new JLabel("Value : ", JLabel.CENTER);
    JTextField textfield = new JTextField("10", 20);
    JButton button = new JButton("Close");
    public void stateChanged(ChangeEvent e) { .. }
    public void keyTyped(KeyEvent e) { .. }
    public void keyPressed(KeyEvent e) { .. }
    public void keyReleased(KeyEvent e) { .. }
    public void actionPerformed(ActionEvent e) { .. }
}
    
```



□ Java2-lab5_1 폴더에 저장 후 제출 (본인 추가 구현 부분)

Lab #5_1 Slider (GUI)



Lab #5_1 ChangeListener, KeyListener, ActionListener Interface

```

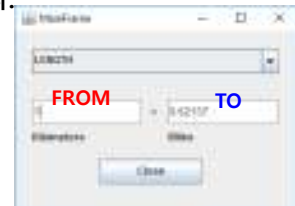
public void stateChanged(ChangeEvent e) {
    textfield.setText("" + ((JSlider)e.getSource()).getValue());
}
public void keyTyped(KeyEvent e) { }
public void keyPressed(KeyEvent e) { }
public void keyReleased(KeyEvent e) {
    int key = e.getKeyCode();
    if (key == KeyEvent.VK_ENTER) {
        Toolkit.getDefaultToolkit().beep();
        System.out.println("ENTER pressed");
        JTextField textfield = (JTextField) e.getSource();
        int value = Integer.parseInt(textfield.getText()); // String -> int
        slider.setValue(value);
    }
}
    
```

Lab #5_2 MetricConverter (OOP & GUI)

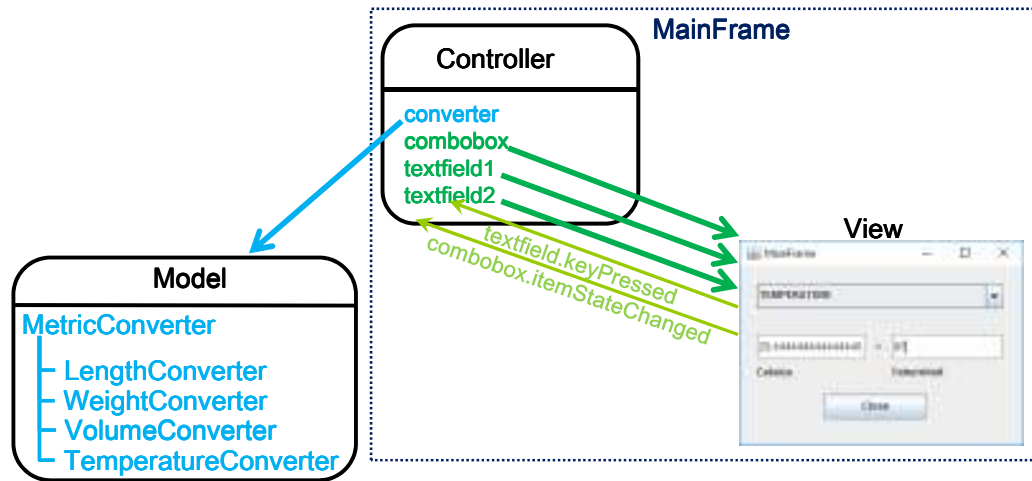
□ Lab#5_2에서는 콤보 박스에서 선택한 측정 기준(LENGTH, WEIGHT, VOLUME, TEMPERATURE)으로 변환하는 MetricConverter를 만든다.

- MetricConverter 인터페이스 FROM <-> TO 변환
- LengthConverter 클래스는 Kilometer <-> Mile 변환
- WeightConverter 클래스는 Kilogram <-> Pound 변환
- VolumeConverter 클래스는 Liter <-> Gallon 변환
- TemperatureConverter 클래스는 Celsius <-> Fahrenheit 변환
- MainFrame 클래스는 JComboBox, JLabel, JTextField, JButton을 이용하여 사용자가 선택한 측정 기준에 따라서 해당 클래스를 호출하여 변환하도록 한다.

□ Java2-lab5_2 폴더에 저장 후 제출



Lab #5_2 MetricConverter (OOP & GUI)

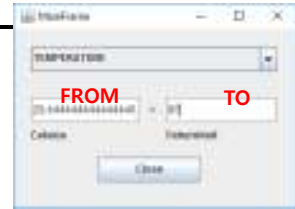


Lab #5_2 MetricConverter (OOP & GUI)

□ MetricConverter 인터페이스

```

public interface MetricConverter {
    double convertFrom(double value); // from -> to
    double convertTo(double value); // to -> from
}
    
```



□ TemperatureConverter 클래스 (Celsius <-> Fahrenheit)

```

public class TemperatureConverter implements MetricConverter {
    public double convertFrom(double value) { return fromCelsiusToFahrenheit(value); }
    public double convertTo(double value) { return fromFahrenheitToCelsius(value); }
    static double fromCelsiusToFahrenheit(double C) { return ((9.0 / 5.0) * C + 32.0); }
    static double fromFahrenheitToCelsius(double F) { return (F - 32.0) * (5.0 / 9.0); }
}
    
```

Lab #5_2 MetricConverter (OOP & GUI)

□ LengthConverter 클래스 (Kilometer <-> Mile)

```

static double fromKilometerToMile(double km) { return (km * 0.62137); }
static double fromMileToKilometer(double mile) { return (mile / 0.62137); }
    
```

□ WeightConverter 클래스 (Kilogram <-> Pound)

```

static double fromKilogramToPound(double kg) { return (kg / 0.45359237); }
static double fromPoundToKilogram(double lbs) { return (lbs * 0.45359237); }
    
```

□ VolumeConverter 클래스 (Liter <-> Gallon)

```

static double fromLiterToGallon(double liter) { return (liter * 0.26417); }
static double fromGallonToLiter(double gallon) { return (gallon / 0.26417); }
    
```

Lab #5_2 MetricConverter (OOP & GUI)

□ MainFrame 클래스는 GUI를 구현한다.

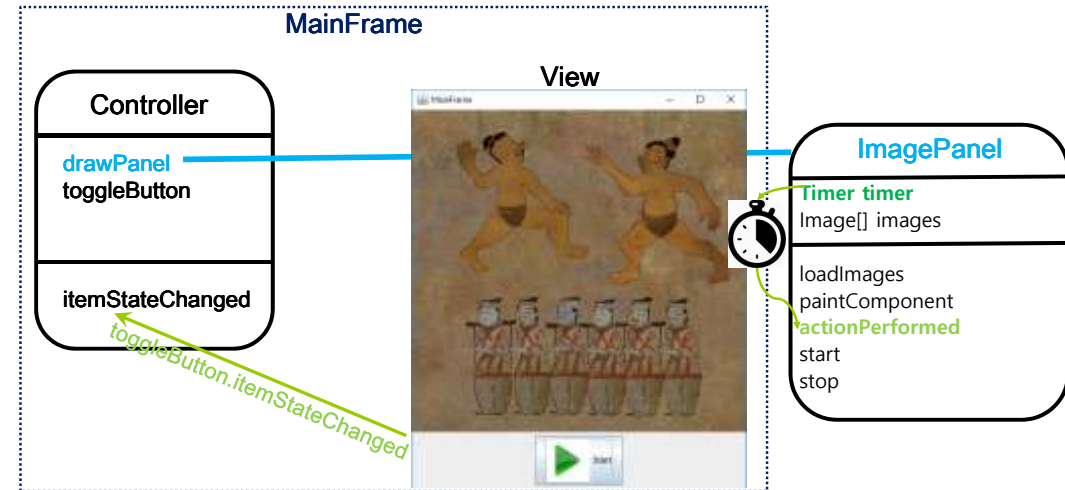
```

public class MainFrame extends JFrame implements ItemListener, KeyListener {
    String[] comboboxItems = {"LENGTH", "WEIGHT", "VOLUME", "TEMPERATURE"};
    JComboBox combobox = new JComboBox(comboboxItems);
    JLabel label1 = new JLabel("Kilometers");
    JLabel label2 = new JLabel("Miles");
    JTextField textfield1 = new JTextField();
    JTextField textfield2 = new JTextField();
    MetricConverter converter = new LengthConverter(); // default converter
    public void itemStateChanged(ItemEvent e) { .. }
    public void keyTyped(KeyEvent e) { .. }
    public void keyPressed(KeyEvent e) { .. }
    public void keyReleased(KeyEvent e) { .. }
}
    
```

Lab #5_3 AnimatedImagePanel (Custom Panel)

- Lab#5_3에서는 C:/JAVA/tacgun1~7.jpg 이미지를 ImagePanel에 로딩하여 start/stop 버튼을 누르면 애니메이션 플레이 또는 정지한다.
 - ImagePanel 클래스는 애니메이션 이미지가 가능한 Panel이다.
 - Image[] images에 애니메이션 이미지를 로딩한다.
 - Timer를 사용하여 타이머 start/restart/stop을 컨트롤한다.
 - 타이머가 동작할 시, 애니메이션이 보이도록 이미지를 바꿔준다.
 - MainFrame 클래스는 ImagePanel과 JToggleButton을 생성하여 애니메이션 start 또는 stop을 할 수 있다.
 - JToggleButton에 play & pause icon image를 넣는다.
 - 그리고 토글 버튼이 SELECTED 되었는지 아니면 DESELECTED되었는지를 판별하여 ImagePanel의 timer를 start/stop 시킨다.
- Java2-lab5_3 폴더에 저장 후 제출

Lab #5_3 AnimatedImagePanel (Custom Panel)



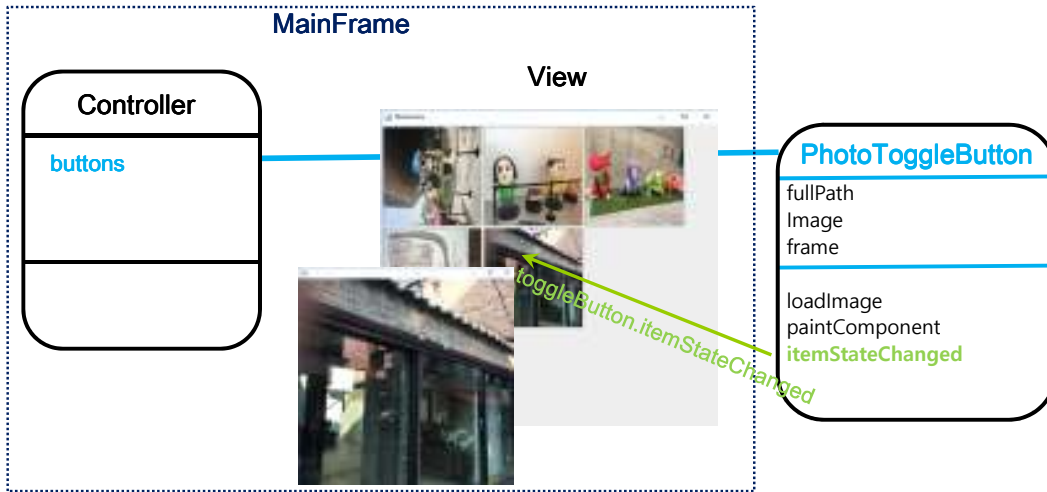
Lab #5_3 AnimatedImagePanel (Custom Panel)

- ImagePanel 클래스
 - `public class ImagePanel extends JPanel implements ActionListener {`
 - `public ImagePanel() {`
 - `loadImages();`
 - `timer = new Timer(500, this);`
 - `}`
 - `public void paintComponent(Graphics g) {`
 - `super.paintComponent(g);`
 - `Graphics2D g2 = (Graphics2D) g;`
 - `if (images[currentImage] != null) g2.drawImage(images[currentImage], 0, 0, this);`
 - `}`
 - `public void actionPerformed(ActionEvent e) { // timer tick`
 - `currentImage = (currentImage + 1) % totalImages;`
 - `repaint();`
 - `}`

Lab #5_4 PhotoToggleButton (Custom ToggleButton)

- Lab#5_4에서는 C:/JAVA 이미지를 토글버튼에 썸네일 이미지로 로딩하고, 토글 버튼을 누르면 큰 사이즈 이미지를 새로운 프레임에 보여주고, 다시 토글 버튼을 누르면 그 프레임을 안보이게 한다.
- Java2-lab5_4 폴더에 저장 후 제출

Lab #5_4 PhotoToggleButton (Custom ToggleButton)



Lab #5_4 PhotoToggleButton (Custom ToggleButton)

□ PhotoToggleButton 클래스

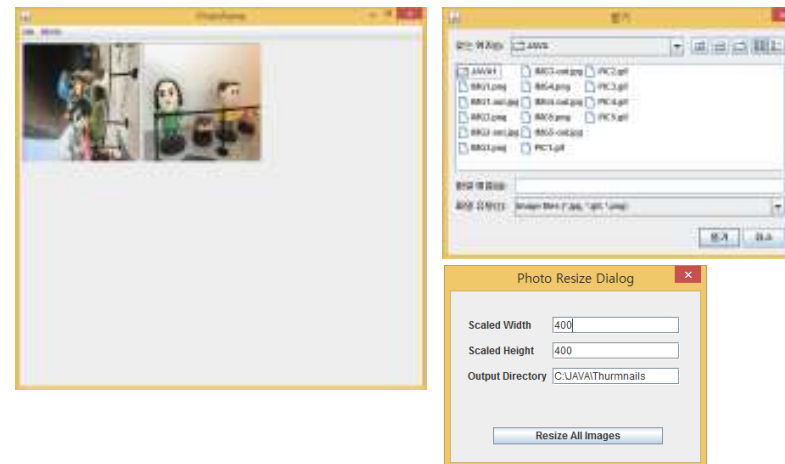
```

public class PhotoToggleButton extends JToggleButton implements ItemListener {
    public PhotoToggleButton(String name, String fullpath) {
        this.image = loadImage(this.fullpath);
        setIcon(loadImageIcon(this.image, 200, 200));
        addItemListener(this);
    }
    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        Graphics2D g2 = (Graphics2D) g;
        if (this.image != null) g2.drawImage(this.image, 0, 0, 200, 200, this);
    }
    public void itemStateChanged(ItemEvent e) { // SELECTED/DESELECTED
        if (e.getStateChange() == ItemEvent.SELECTED) ... // display new frame
        else if (e.getStateChange() == ItemEvent.DESELECTED) ... // dispose frame
    }
}
    
```

Lab #5_5 PhotoManager (Menu & Dialog)

- Lab#5_5에서는 Lab#5_4를 확장해서 메뉴와 ImageConverter/ImageResizer Dialog를 추가한다.
- 사용자가 선택한 이미지들을 가지고, SlideShow 기능을 추가해본다.
- Java2-lab5_5 폴더에 저장 후 제출

Lab #5_5 PhotoManager (Menu & Dialog)



과제 제출

- Lab05_1 ~ Lab05_5와 보고서를 전체적으로 묶어서 e-learning에 과제 제출
- 각 Lab마다 **본인이 추가로 작성한 코드**와 설명을 중점적으로 보고할 것!