

20장. GUI 프로그래밍



스윙으로 윈도우 만들기



목 차

1

윈도우의 구성 요소

2

레이아웃과 패널

3

버튼 이벤트 - ActionListener

4

응용 프로젝트 - 계산기와 환율변환기

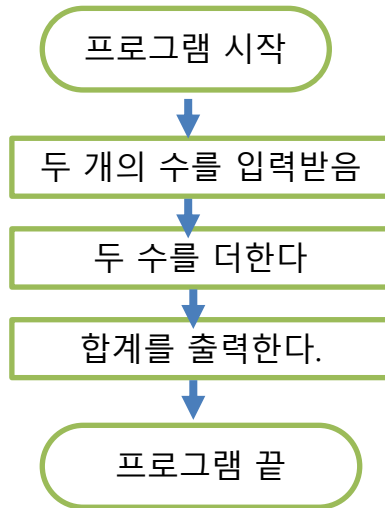


GUI 프로그래밍

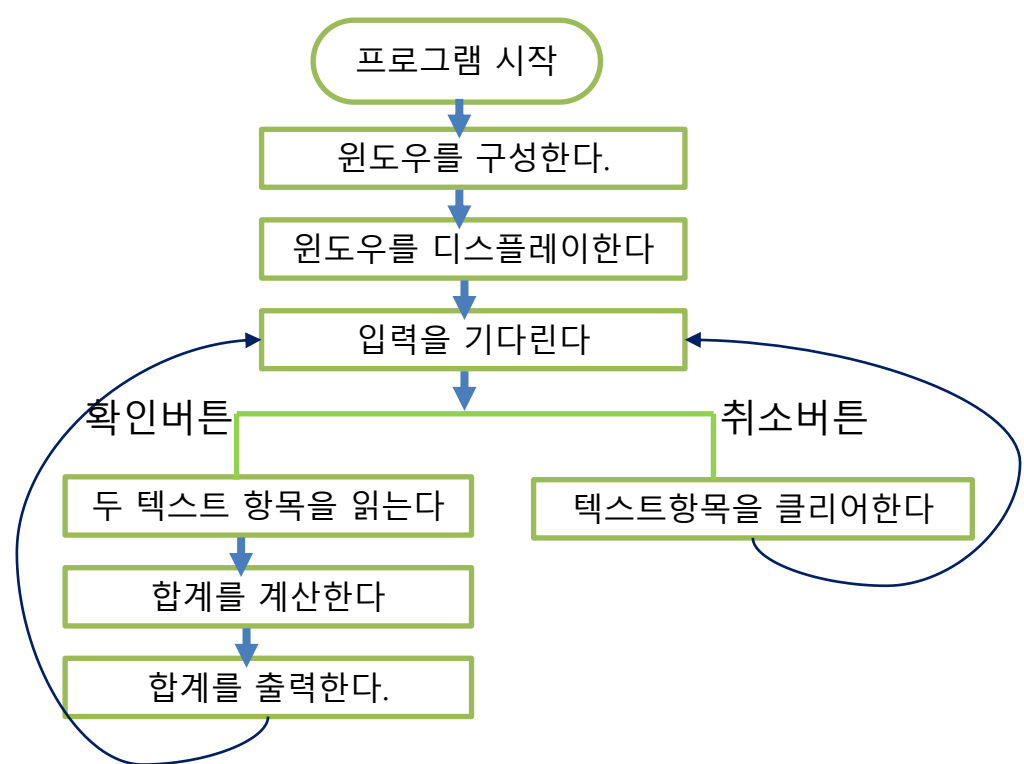
■ GUI 환경에서 작동하는 프로그램

GUI란 윈도우, 아이콘, 메뉴와 같은 시각적인 요소들을 통해 프로그램을 실행하는 환경을 말한다.

<텍스트 모드 프로그램의 구조>



<윈도우 프로그램의 구조>

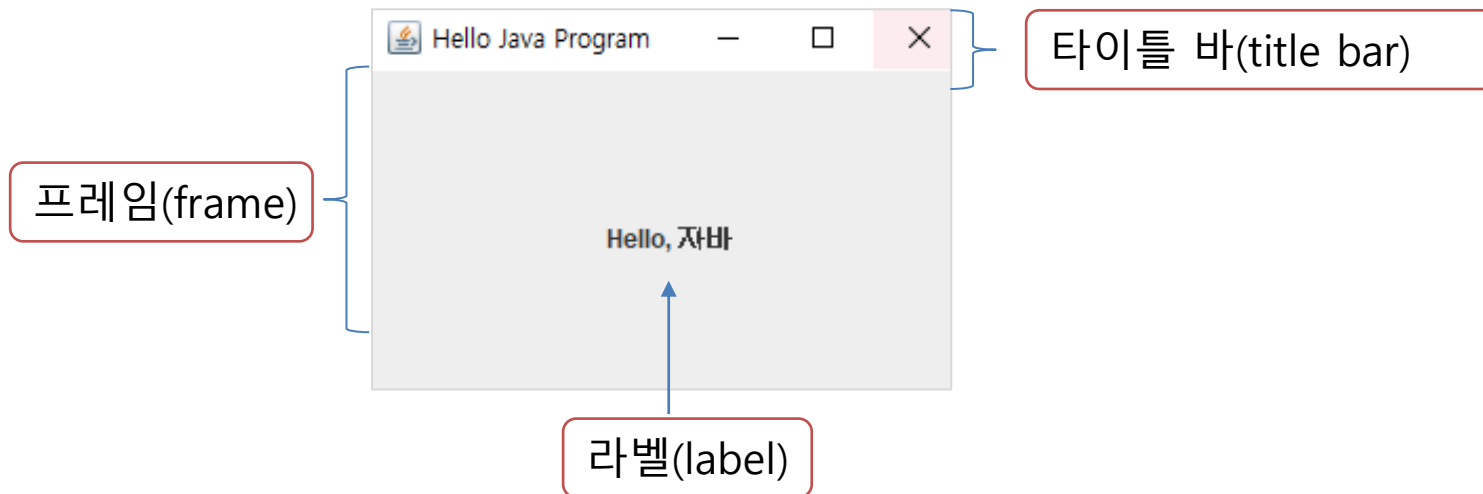


GUI 프로그래밍

■ 윈도우의 구성 요소

윈도우를 구성하는 모든 요소들을 컴포넌트(component)라고 한다.

- 프레임(frame) : 윈도우 전체
- 패널(panel) : 프로그램에 필요한 데이터를 입력 받거나, 실행 결과를 출력하는 부분
- 라벨(Label) : 텍스트를 출력(표시)하는 컴포넌트이다.



GUI 프로그래밍

■ 윈도우를 구성하는 방법

윈도우 프로그램을 만드는 데 사용되는 클래스는 JDK의 java.awt 패키지와 javax.swing 패키지 안에 있다.

- java.awt 패키지 : 최초 버전부터 사용되어온 패키지
- javax.swing 패키지 : 성능과 플랫폼 중립성을 고려하여 나중에 추가된 패키지

①

```
JFrame frame = new JFrame("Hello Program")
```

②

```
JLabel label = new JLabel("Hello, Java")
```

③

```
frame.add(label);
```



- 윈도우를 디스플레이하는 방법

프레임을 디스플레이하면 그 위에 있는 컴포넌트들도 자동으로 디스플레이 된다.

①

```
frame.setSize(200, 200) //크기
```

②

```
frame.Location(300, 300) //위치
```

③

```
frame.add(label)
```

④

```
frame.setVisibel(true)
```

⑤

```
frame.setDefaultCloseOperation(Jframe.EXIT_ON_CLOSE)
```



GUI 프로그래밍

▪ Swing 컨테이너

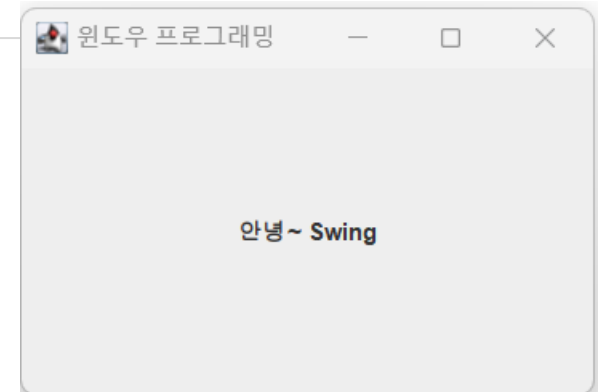
컨테이너 클래스	용 도
JFrame	작업표시줄, 메뉴가 제공되는 윈도우를 만들때 사용
JPanel	컴포넌트들을 배치할 때 사용
JLabel	출력을 위한 레이블을 만들 때 사용
JButton	버튼을 만들어 버튼 이벤트를 만들때 사용
TextField	글자를 입력할 수 있는 입력 상자를 만들때 사용



GUI 프로그래밍

■ JFrame > JLabel

```
public class JFrame1 {  
  
    public static void main(String[] args) {  
        //JFrame frame = new JFrame("윈도우 프로그래밍");  
        JFrame frame = new JFrame();  
        frame.setTitle("윈도우 프로그래밍");  
        frame.setSize(300, 200);        //width, height  
        frame.setLocation(100, 100);    //x, y 좌표  
        //frame.setBounds(100, 100, 300, 200); //x, y, width, height  
  
        JLabel label = new JLabel("안녕~ Swing");  
        label.setHorizontalAlignment(JLabel.CENTER);  
        frame.add(label);  
  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
  
        frame.setVisible(true);  
    }  
}
```



▪ Layout Manager

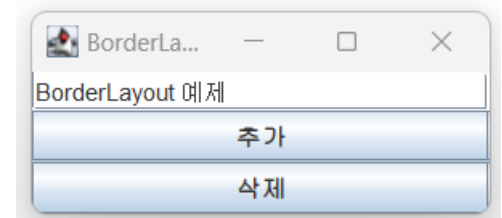
컨테이너 클래스	용 도
BorderLayout	컴포넌트를 동, 서, 남, 북, 중앙으로 배치
FlowLayout	컴포넌트를 왼쪽에서 오른쪽으로 일렬로 배치
GridLayout	컴포넌트를 바둑판과 같은 격자에 배치



GUI 프로그래밍

■ JTextField & JButton – BorderLayout 예제

```
public class BorderLayoutExample {  
  
    public static void main(String[] args) {  
        JFrame frame = new JFrame();  
        frame.setTitle("BorderLayout");  
        frame.setSize(250, 110);    //width, height  
        frame.setLocation(100, 100); //x, y 좌표  
  
        JTextField txt = new JTextField("BorderLayout 예제");  
        JButton btn1 = new JButton("추가");  
        JButton btn2 = new JButton("삭제");  
  
        //컴포넌트 배치  
        frame.add(txt, BorderLayout.NORTH); //북쪽  
        frame.add(btn1, BorderLayout.CENTER); //동쪽  
        frame.add(btn2, BorderLayout.SOUTH); //남쪽  
  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setVisible(true);  
    }  
}
```



GUI 프로그래밍

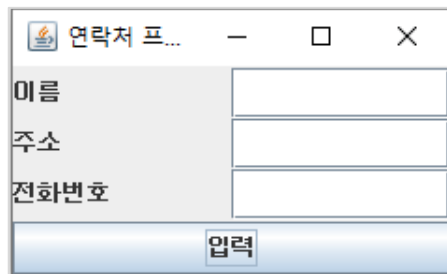
■ FlowLayout 예제

```
public class FlowLayoutExample {  
  
    public static void main(String[] args) {  
        JFrame frame = new JFrame();  
        frame.setTitle("FlowLayout");  
        frame.setSize(500, 100);  
        frame.setLocation(100, 100);  
  
        FlowLayout flowLayout = new FlowLayout();  
        frame.setLayout(flowLayout);  
  
        //컴포넌트 배치  
        frame.add(new JButton("계란"));  
        frame.add(new JButton("커피"));  
        frame.add(new JButton("두부"));  
        frame.add(new JButton("콩나물"));  
        frame.add(new JButton("쌀"));  
  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setVisible(true);  
    }  
}
```



GUI 프로그래밍

■ GridLayout 예제



```
public class GridLayoutFrame extends JFrame{

    public GridLayoutFrame() {
        setTitle("연락처 프로그램");
        setSize(250, 150);
        setLocation(500, 400);
        Container contentPane = getContentPane();

        //contentPane의 "Center" - JPanel배치
        //contentPane의 "South" - Button 배치
        JPanel panel = new JPanel();
        panel.setLayout(new GridLayout(3, 2));
        //layout을 GridLayout으로 설정

        JTextField text1 = new JTextField();
        JTextField text2 = new JTextField();
        JTextField text3 = new JTextField();
```



GUI 프로그래밍

GridLayout

```
panel.add(new JLabel("이름"));
panel.add(text1);
panel.add(new JLabel("주소"));
panel.add(text2);
panel.add(new JLabel("전화번호"));
panel.add(text3);

contentPane.add(panel, "Center");
contentPane.add(new JButton("입력"), "South");

setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public static void main(String[] args) {
    new GridLayoutFrame();
}
```

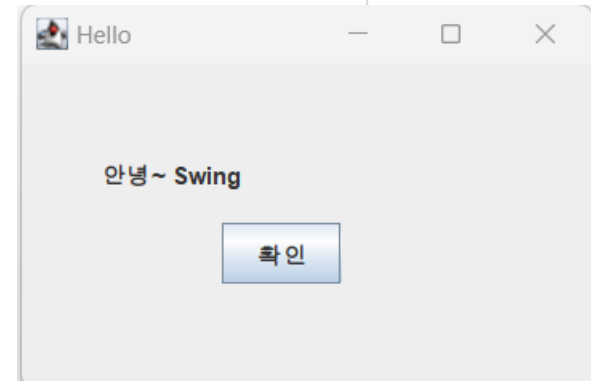
BorderLayout 사용



GUI 프로그래밍

- JFrame를 상속받아 윈도우 만들기

```
public class MyFrame extends JFrame{  
  
    public MyFrame() { //생성자  
        setTitle("Hello");  
        setSize(300, 200);  
        setLocation(100, 100);  
        setLayout(null);  
  
        //레이블 생성  
        JLabel lbl = new JLabel("안녕~ Swing");  
        lbl.setSize(80, 30);  
        lbl.setLocation(40, 40);  
        add(lbl);  
  
        //버튼 생성  
        JButton btn = new JButton("확인");  
        btn.setSize(60, 30);  
        btn.setLocation(100, 80);  
        add(btn);  
    }  
}
```



GUI 프로그래밍

- JFrame를 상속받아 윈도우 만들기

```
//버튼 생성
JButton btn = new JButton("확인");
btn.setSize(60, 30);
btn.setLocation(100, 80);
add(btn);

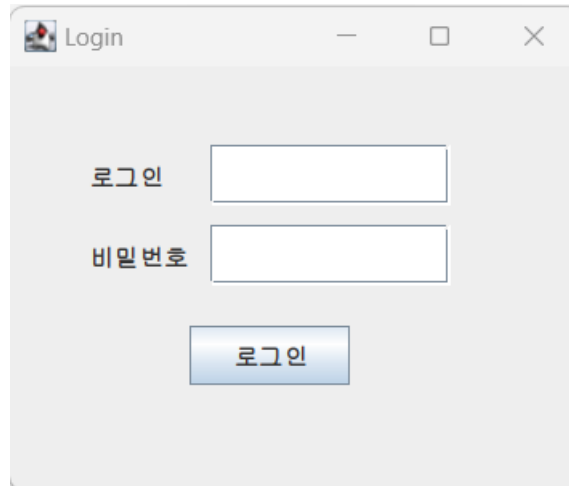
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //종료
setVisible(true); //화면 출력
}

public static void main(String[] args) {
    //객체 생성시 윈도우 생성
    //MyFrame frame = new MyFrame();
    new MyFrame();
}
}
```



실습 예제 – 로그인 창

- 로그인 윈도우



Login

로그인

비밀 번호

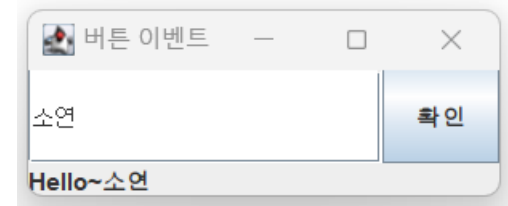
로그인



GUI 프로그래밍

■ 버튼 이벤트 예제

```
public class ButtonActionEvent {  
  
    public static void main(String[] args) {  
        JFrame frame = new JFrame();  
        frame.setTitle("버튼 이벤트");  
        frame.setSize(250, 100);    //width, height  
        frame.setLocation(100, 100); //x, y 좌표  
  
        JTextField txt = new JTextField();  
        JButton btn = new JButton("확인");  
        JLabel lbl = new JLabel("Hello");  
  
        //button 이벤트 - ConfirmBtnActionListener 클래스 호출  
        ConfirmBtnActionListener listener =  
            new ConfirmBtnActionListener(txt, lbl);  
        btn.addActionListener(listener);  
  
        //컴포넌트 배치 - BorderLayout.CENTER은 "Center"와 같음  
        frame.add(txt, "Center"); //중앙  
        frame.add(btn, "East");    //동쪽  
        frame.add(lbl, "South");   //남쪽  
    }  
}
```



addActionListener() 메서드



GUI 프로그래밍

ConfirmBtnActionListener 클래스 만들기

```
public class ConfirmBtnActionListener implements ActionListener{
    //ActionListener 구현 클래스 정의
    JTextField text;
    JLabel label;

    public ConfirmBtnActionListener(JTextField text, JLabel label) {
        this.text = text;
        this.label = label;
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String name = text.getText(); // 입력 상자의 내용 가져오기
        label.setText("Hello " + name); // name을 라벨에 세팅하기
    }
}
```

매개변수 있는 생성자



GUI 프로그래밍

버튼(Button) 이벤트 예제 - 익명 객체로 구현

```
public class ButtonActionEvent2 extends JFrame{  
  
    public ButtonActionEvent2() {  
        setTitle("버튼 이벤트");  
        setSize(250, 100);    //width, height  
        setLocation(100, 100); //x, y 좌표  
  
        JTextField txt = new JTextField();  
        JButton btn = new JButton("확인");  
        JLabel lbl = new JLabel("Hello");  
  
        //button 이벤트 - 익명 객체로 구현  
        ActionListener listener = new ActionListener() {  
  
            @Override  
            public void actionPerformed(ActionEvent e) {  
                String name = txt.getText();  
                lbl.setText("Hello~" + name);  
            }  
        };  
        btn.addActionListener(listener);  
    }  
}
```



GUI 프로그래밍

버튼(Button) 이벤트 예제 - 익명 객체로 구현

```
//컴포넌트 배치
add(txt, "Center"); //중앙
add(btn, "East");   //동쪽
add(lbl, "South");  //남쪽

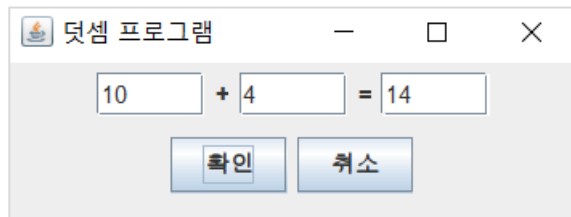
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setVisible(true);
}

public static void main(String[] args) {
    new ButtonActionEvent2();
}
}
```



덧셈 계산 프로그램

덧셈 계산 프로그램



```
public class plusCalc {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("덧셈 프로그램");  
        frame.setSize(250, 120);  
        frame.setLocation(400, 400);  
  
        Container contentPane = frame.getContentPane();  
  
        JPanel pane1 = new JPanel();  
        JPanel pane2 = new JPanel();  
  
        contentPane.add(pane1, "North");  
        contentPane.add(pane2, "Center");  
  
        //패널1 배치  
        pane1.setLayout(new FlowLayout());  
        JTextField text1 = new JTextField(5);  
        JTextField text2 = new JTextField(5);  
        JTextField text3 = new JTextField(5);  
  
        //패널2 배치  
        pane2.setLayout(new FlowLayout());  
        JButton btn1 = new JButton("확인");  
        JButton btn2 = new JButton("취소");
```



덧셈 계산 프로그램

덧셈 계산 프로그램

```
//패널1에 컴포넌트 올리기
pane1.add(text1);
pane1.add(new JLabel("+"));
pane1.add(text2);
pane1.add(new JLabel("="));
pane1.add(text3);
```

```
//패널2에 컴포넌트 올리기
pane2.add(btn1);
pane2.add(btn2);
```

```
ActionListener listener = new ActionListener() {
    //확인 이벤트 처리
    @Override
    public void actionPerformed(ActionEvent e) {
        int num1 = Integer.parseInt(text1.getText());
        int num2 = Integer.parseInt(text2.getText());
        int sum = num1 + num2;
        text3.setText(String.valueOf(sum));
    }
};
btn1.addActionListener(listener);
```

JFrame

pane1

JTextField

pane2

JButton

숫자를 문자로 변환



덧셈 계산 프로그램

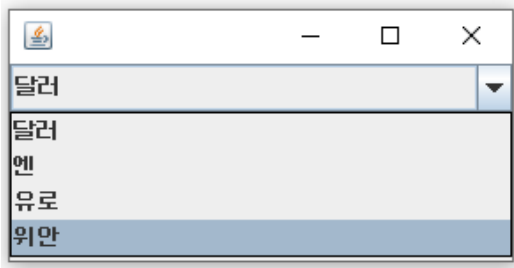
덧셈 계산 프로그램

```
    ActionListener listener2 = new ActionListener() {  
        //취소 이벤트 처리  
        @Override  
        public void actionPerformed(ActionEvent e) {  
            text1.setText("");  
            text2.setText("");  
            text3.setText("");  
        }  
    };  
    btn2.addActionListener(listener2);  
  
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    frame.setVisible(true);  
}
```



콤보 박스(ComboBox)

ComboBox로 목록 메뉴를 선택하는 프로그램 만들기



```
String[] money = {"달러", "엔", "유로", "위안"};
```

```
JComboBox<String> comboBox;
```

생성자의 매개변수

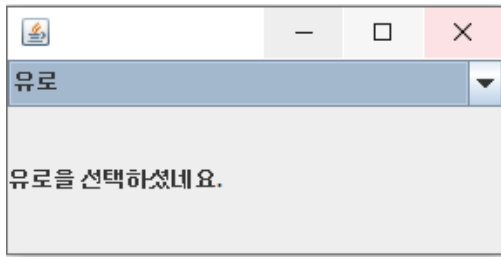
```
comboBox = new JComboBox<>(money);
```

```
label = new JLabel("화폐를 선택해 주세요");
```



콤보 박스(ComboBox)

ComboBox로 목록 메뉴를 선택하는 프로그램 만들기



```
ActionListener listener = new ActionListener() {  
    //콤보 상자를 선택하면 라벨에 화폐 출력  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        String money = (String) comboBox.getSelectedItem();  
        label.setText(money + "를 선택하셨습니다.");  
    }  
};  
comboBox.addActionListener(listener);
```



콤보 박스(ComboBox)

```
public class ComboBoxExample extends JFrame{

    String[] money = {"달러", "엔", "유로", "위안"};
    JComboBox<String> comboBox;
    JLabel label;

    public ComboBoxExample() {
        setTitle("목록 메뉴 선택");
        setLocation(300, 300);
        setSize(300, 150);
        Container contentPane = getContentPane();

        //컴포넌트 객체 생성
        comboBox = new JComboBox<>(money);
        label = new JLabel("화폐를 선택해 주세요");

        //컴포넌트 배치
        contentPane.add(comboBox, "North");
        contentPane.add(label, "Center");
    }
}
```

```
    ActionListener listener = new ActionListener() {
        //콤보 상자를 선택하면 라벨에 화폐 출력
        @Override
        public void actionPerformed(ActionEvent e) {
            String money = (String) comboBox.getSelectedItem();
            label.setText(money + "를 선택하셨습니다.");
        }
    };
    comboBox.addActionListener(listener);

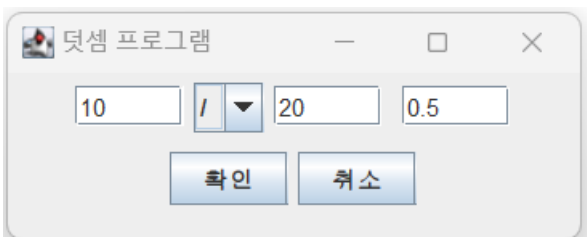
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public static void main(String[] args) {
    new ComboBoxExample();
}
```



사칙연산 계산기

사칙연산 계산기



Module `java.desktop`

Package `javax.swing`

Class `JComboBox<E>`

`java.lang.Object`
`java.awt.Component`
`java.awt.Container`
`javax.swing.JComponent`
`javax.swing.JComboBox<E>`

Type Parameters:

E - the type of the elements of this combo box

All Implemented Interfaces:

`ActionListener`, `ImageObserver`, `ItemSelectable`, `M`

```
@JavaBean(defaultProperty="UI",
           description="A combination of a text
public class JComboBox<E>
extends JComponent
implements ItemSelectable, ListDataListener,
```



GUI 프로그래밍

사칙연산 계산기

콤보박스과 사칙연산자

```
String[] operator = {"+", "-", "*", "/"};  
JComboBox<String> comboBox = new JComboBox<>(operator);
```

```
ActionListener listener = new ActionListener() {  
    @Override //확인 이벤트 처리  
    public void actionPerformed(ActionEvent e) {  
        String operator = (String) comboBox.getSelectedItem();  
        int num1 = Integer.parseInt(text1.getText());  
        int num2 = Integer.parseInt(text2.getText());  
        if(operator.equals("+")) {  
            int result = num1 + num2;  
            text3.setText(String.valueOf(result));  
        }  
    }  
};
```

반환값이 Object -> String



사칙연산 계산기

사칙연산 계산기

```
public class Calculator extends JFrame{

    public Calculator() {
        setTitle("덧셈 프로그램");
        setSize(300, 120);
        setLocation(100, 100);

        //패널 2개 생성
        JPanel pane1 = new JPanel();
        JPanel pane2 = new JPanel();

        add(pane1, BorderLayout.NORTH);
        add(pane2, BorderLayout.CENTER);

        //패널 배치
        pane1.setLayout(new FlowLayout());
        JTextField text1 = new JTextField(5);
        //연산자
        String[] operator = {"+", "-", "*", "/"};
        JComboBox<String> comboBox = new JComboBox<>(operator);
```



사칙연산 계산기

사칙연산 계산기

```

JTextField text2 = new JTextField(5);
JTextField text3 = new JTextField(5);
JLabel label = new JLabel();

pane2.setLayout(new FlowLayout());
JButton confirmBtn = new JButton("확인");
JButton cancelBtn = new JButton("취소");

//pane에 컴포넌트 올리기
pane1.add(text1);
pane1.add(comboBox);
pane1.add(text2);
pane1.add(label);
pane1.add(text3);

pane2.add(confirmBtn);
pane2.add(cancelBtn);

```



사칙연산 계산기

```

ActionListener listener = new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int num1 = Integer.parseInt(text1.getText());
        String op = (String)comboBox.getSelectedItem();
        int num2 = Integer.parseInt(text2.getText());

        switch(op) {
            case "+":
                int result = num1 + num2;
                text3.setText(String.valueOf(result)); break;
            case "-":
                result = num1 - num2;
                text3.setText(String.valueOf(result)); break;
            case "*":
                result = num1 * num2;
                text3.setText(String.valueOf(result)); break;
            case "/":
                double result2 = (double)num1 / num2;
                text3.setText(String.valueOf(result2)); break;
        }
    }
};
confirmBtn.addActionListener(listener);

```



사칙연산 계산기

```
//취소
ActionListener listener2 = new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        text1.setText("");
        text2.setText("");
        text3.setText("");
    }
};
cancelBtn.addActionListener(listener2);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setVisible(true);
}

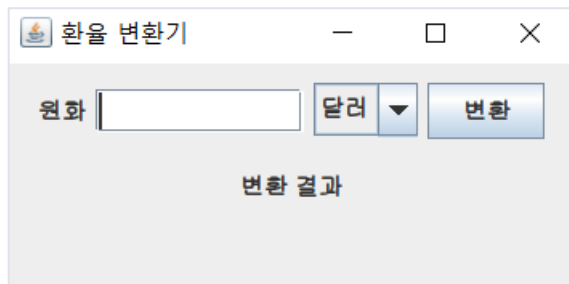
public static void main(String[] args) {
    new Calculator();
}
}
```



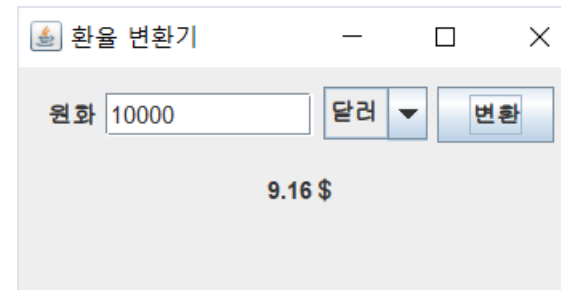
환율 변환기

환율 변환기 프로그램

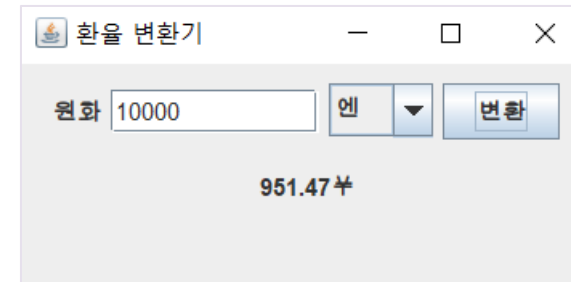
원화를 입력하고 통화를 선택한 후 변환 버튼을 누르면 환전되는 프로그램



The initial state of the '환율 변환기' (Currency Converter) window. It features a title bar with the program name and standard window controls. The main area contains a text input field labeled '원화' (Korean Won) which is currently empty. To the right of the input field is a dropdown menu showing '달러' (Dollar) and a '변환' (Convert) button. Below these elements, the text '변환 결과' (Conversion Result) is displayed.



The first converted state of the '환율 변환기' window. The '원화' input field now contains the value '10000'. The dropdown menu remains set to '달러'. The '변환' button is present. The result '9.16 \$' is displayed at the bottom of the window.



The second converted state of the '환율 변환기' window. The '원화' input field contains '10000'. The dropdown menu has been changed to '엔' (Yen). The '변환' button is present. The result '951.47 ₩' is displayed at the bottom of the window.



환율 변환기

환율 변환기

```
public class Exchange extends JFrame{
    //환율 변환기 - 2021-1-11 기준 환율
    private static final float USD = 1092.00F; //달러
    private static final float JPY = 10.51F; //엔
    private static final float EUR = 1336.23F; //유로
    private static final float CNY = 169.07F; //위안

    String[] currency = {"달러", "엔", "유로", "위안"};

    JPanel panel1;
    JPanel pane2;
    JTextField text;
    JComboBox<String> combo;
    JButton button;
    JLabel label;

    public Exchange() {
        setTitle("환율 변환기");
        setLocation(800, 200);
        setSize(300, 150);
        Container contentPane = getContentPane();

        panel1 = new JPanel();
        pane2 = new JPanel();
        combo = new JComboBox<>(currency);
        text = new JTextField(10);
        button = new JButton("변환");
        label = new JLabel("변환 결과");
    }
}
```



환율 변환기

환율 변환기

```
//contentPane 레이아웃(배치)
contentPane.add(pane1, "North");
contentPane.add(pane2, "South");

//pane1 레이아웃 - 기본 레이아웃을 FlowLayout으로 변경
contentPane.setLayout(new FlowLayout());
pane1.add(new JLabel("원화"));
pane1.add(text);
pane1.add(combo);
pane1.add(button);

//pane2 레이아웃
contentPane.setLayout(new FlowLayout());
pane2.add(label);

ActionListener listener = new ActionListener() {
    //변환 버튼을 누르면 환율 계산
    @Override
    public void actionPerformed(ActionEvent e) {
        calculate();
    }
};
button.addActionListener(listener);

setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
```



환율 변환기

환율 변환기

```
private void calculate() {
    float won = Float.parseFloat(text.getText());
    String currency = (String) combo.getSelectedItem();
    if(currency.equals("달러")) {
        won = won / USD;
        //label.setText(String.valueOf(won));
        label.setText(String.format("%.2f$", won));
    }else if(currency.equals("엔")) {
        won = won / JPY;
        label.setText(String.format("%.2f¥", won));
    }else if(currency.equals("유로")) {
        won = won / EUR;
        label.setText(String.format("%.2f€", won));
    }else if(currency.equals("위안")) {
        won = won / CNY;
        label.setText(String.format("%.2f¥", won));
    }
}

public static void main(String[] args) {
    new Exchange();
}
```

