Lecture 3

GUI: 배치관리자

2018년도 2학기

컴퓨터프로그래밍2

김 영 국 충남대학교 컴퓨터공학과





이번 주에 학습할 내용



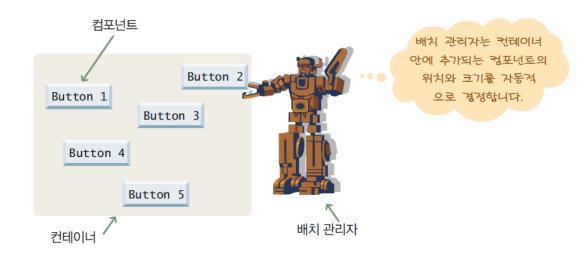
- 배치관리자의 개요
- 배치관리자의 사용
- FlowLayout
- BorderLayout
- GridLayout
- BoxLayout
- CardLayout
- 절대위치로 배치

컨테이너 안에서 컴포넌트를 배치하는 방법에 대하여 살펴봅시다.





 컨테이너 안의 각 컴포넌트의 위치와 크 기를 결정하는 작업



< 배치관리자의 개념 >



FlowLayout



BorderLayout



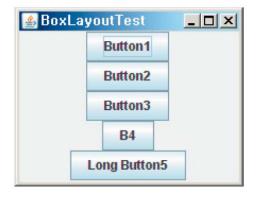
GridLayout





배치관리자의 개요

BoxLayout



CardLayout





배치관리자를 사용하는 방법

■ 크기와 정렬 힌트 제공하기

```
panel.setLayout(new BorderLayout()); // 패널의 배치 관리자를 설정한다.
button.setMaximumSize(new Dimension(300, 200)); // 최대 크기 힌트
button.setAlignmentX(JComponent.CENTER_ALIGNMENT); // 중앙 정렬 힌트
```

■ 배치 방향 설정

panel.applyComponentOrientation(ComponentOrientation.RIGHT_TO_LEFT);





FlowLayout 클래스

생성자	설명
FlowLayout()	새로운 FlowLayout 객체를 생성한다. 기본 설정은 중앙(center) 배치이며 간격은 세로, 가로 각각 5 픽셀이다.
FlowLayout(int align)	지정된 정렬 방식을 가진 새로운 FlowLayout 객체를 생성한다. 기본 설정은 중앙(center) 배치이며 간격은 세로, 가로 각각 5 픽셀이다. 정렬 매개변수는 다음 중 하나이다. FlowLayout.LEADING, FlowLayout.TRAILING.
FlowLayout (int align, int hgap, int vgap)	지정된 정렬 방식과 수평 간격 hgap과 수직 간격 vgap을 가진 새로운 FlowLayout 객체를 생성한다.

FlowLayout 예제

```
FlowTest.java
     import java.awt.*;
 01
     import javax.swing.*;
 02
 03
 04
     class MyFrame extends JFrame {
        public MyFrame() {
 05
 06
           setTitle("FlowLayoutTest");
 07
           setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 08
 09
                                                               패널의 배치 관리자를
           JPanel panel;
 10
                                                               FlowLauout으로 설정
           // 패널을 생성하고 배치 관리자를 FlowLayout으로 설정
 11
           panel = new JPanel();
 12
           panel.setLayout(new FlowLayout(FlowLayout.CENTER));
 13
           // 패널에 버튼을 생성하여 추가
 14
           panel.add(new JButton("Button1"));
 15
 16
           panel.add(new JButton("Button2"));
           panel.add(new JButton("Button3"));
                                                  .---- 패널에 버튼 추가
 17
           panel.add(new JButton("B4"));
 18
           panel.add(new JButton("Long Button5"));
 19
           add(panel); ◀------ 패널을 프레임에 추가
 20
```

FlowLayout 예제

```
21
22
          pack();
           setVisible(true);
23
       }
24
25
26
27
    public class FlowTest {
28
       public static void main(String argv[]) {
          MyFrame f = new MyFrame();
29
30
       }
31 }
```

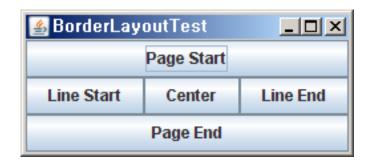
실행결과





BorderLayout 클래스

■ BorderLayout은 5개의 영역으로 구분하고 각 각의 영역에 컴포넌트를 배치



PAGE_START (또는 NORTH)
PAGE_END (또는 SOUTH)
LINE_START (또는 WEST)
LINE_END (또는 EAST)
CENTER

panel.add(aComponent, BorderLayout.*PAGE_START*); // 북쪽에 배치한다.



생성자 또는 메소드	설명
BorderLayout(int hgap, int vgap)	컴포넌트 사이의 수평 간격 hgap과 수직 간격 vgap을 가지는 BorderLayout 객체 생성
setHgap(int)	컴포넌트 사이의 수평 간격 설정(단위는 픽셀)
setVgap(int)	컴포넌트 사이의 수직 간격 설정



```
BorderTest.jav
     import java.awt.*;
 01
     import javax.swing.*;
 02
 03
     class MyFrame extends JFrame {
 04
 05
        public MyFrame() {
 06
           setTitle("BorderLayoutTest");
 07
           setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 08
 09
           // 프레임은 디폴트로 BorderLayout 이므로 사실은 불필요
 10
                                                                프레임에 버튼을 추가한다.
                                                                추가하는 위치를 지정한다.
 11
           setLayout(new BorderLayout());
 12
           // 버튼을 추가한다.
 13
           add(new JButton("Center"), BorderLayout.CENTER);
 14
 15
           add(new JButton("Line Start"), BorderLayout.LINE_START);
 16
           add(new JButton("Line End"), BorderLayout.LINE_END);
 17
           add(new JButton("Page Start"), BorderLayout.PAGE_START);
 18
           add(new JButton("Page End"), BorderLayout.PAGE_END);
```

예제

```
19
20
          pack();
          setVisible(true);
21
22
      }
23
   }
24
    public class BorderTest {
25
       public static void main(String argv[]) {
26
27
          MyFrame f = new MyFrame();
28
       }
29 }
```

실행결과





GridLayout 클래스

■생성자

생성자	설명
GridLayout(int rows, int cols)	rows 행과 cols 열을 가지는 GridLayout 객체를 생성한다. 만약 rows나 cols가 0이면 필요한 만큼의 행이나 열이 만들어진다.
GridLayout(int rows, int cols, int hgap, int vgap)	rows 행과 cols 열을 가지는 GridLayout 객체를 생성한다. hgap과 vgap은 컴포넌트 사이의 수평 간격과 수직 간격으로 단위는 픽셀이다.

setLayout(new GridLayout (3,0));

예제

```
GridTest.java
     import java.awt.*;
 01
     import javax.swing.*;
 02
 03
 04
     class MyFrame extends JFrame {
        public MyFrame() {
 05
 06
            setTitle("GridLayoutTest");
 07
            setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 08
 09
           setLayout(new GridLayout(0, 3)); // 3개의 열과 필요한 만큼의 행
 10
 11
            add(new JButton("Button1"));
 12
            add(new JButton("Button2"));
 13
            add(new JButton("Button3"));
 14
 15
            add(new JButton("B4"));
                                                                        3개의 열을 가진 GridLayout, 행의 개수는
            add(new JButton("Long Button5"));
 16
                                                                        추가되는 컴포넌트에 의하여 결정된다.
 17
            pack();
 18
 19
            setVisible(true);
 20
        }
 21
     }
 22
     public class GridTest {
 23
 24
        public static void main(String argv[]) {
            MyFrame f = new MyFrame();
 25
 26
 27
```



실행결과



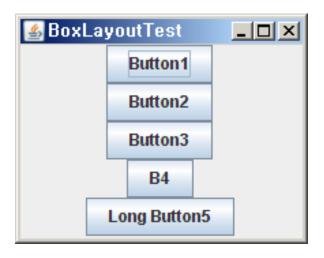
BoxLayout 클래스 예제

```
BoxTest.java
     import java.awt.*;
     import java.awt.event.*;
 02
     import javax.swing.*;
 03
 04
 05
     class MyFrame extends JFrame {
 06
        public MyFrame() {
 07
            setTitle("BoxLayoutTest");
 08
            setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 09
 10
            JPanel panel = new JPanel();
 11
           // Y축 방향으로 컴포넌트를 쌓는다.
 12
            panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));
 13
 14
```

BoxLayout 예제

```
15
          makeButton(panel, "Button1");
          makeButton(panel, "Button2");
16
          makeButton(panel, "Button3");
17
          makeButton(panel, "B4");
18
          makeButton(panel, "Long Button5");
19
           add(panel);
20
           pack();
21
           setVisible(true);
22
23
24
       private void makeButton(JPanel panel, String text) {
25
           JButton button = new JButton(text);
26
           button.setAlignmentX(Component.CENTER_ALIGNMENT);
27
           panel.add(button);
28
29
30
31
    public class BoxTest {
32
                                                                   BoxLayout의 중앙에
       public static void main(String args[]) {
33
                                                                   버튼들을 추가한다.
          MyFrame f = new MyFrame();
34
35
36
```







CardLayout 클래스

■생성자

생성자	설명
CardLayout()	디폴트 생성자
CardLayout(int hgap, int vgap)	hgap과 vgap은 컴포넌트 사이의 수평 간격과 수직 간격을 픽셀로 표시한다. 기본값은 0이다.

■ 메소드

메소드	설명
first (Container parent)	첫 번째 카드를 선택한다.
next (Container parent)	다음 카드를 선택한다. 만약 현재 선택된 카드가 마지막이면 처음으로 간다.
previous (Container parent)	이전 카드를 선택한다.
last (Container parent)	마지막 카드를 선택한다.
show (Container parent, String s)	문자열 s로 지정한 카드를 선택 선택한다.

CardLayout 예제

```
CardTest.java
     import java.awt.*;
    import java.awt.event.*;
 02
    import javax.swing.*;
 03
 04
     class MyFrame extends JFrame implements ActionListener {
 05
        JPanel panel;
 06
        Cards cards:
 07
 08
 09
        public MyFrame() {
                                                                GridLauout의 패널을 만들고
            setTitle("CardLayoutTest");
 10
                                                                여기에 계어 버튼을 넣는다.
 11
            setSize(400, 200);
 12
            setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
           panel = new JPanel(new GridLayout(0, 5, 10, 0));
 13
           addButton("<<", panel);</pre>
 14
           addButton("<", panel);</pre>
 15
           addButton(">", panel);
 16
 17
            addButton(">>", panel);
 18
           addButton("종료", panel);
 19
           add(panel, "North");
           cards = new Cards();
 20
           add(cards, "Center");
 21
                                                                Cards 객체를 생성하여 중앙에 배치한다.
 22
           setVisible(true);
                                                                Cards는 패널의 일종이다(뒤에 정의된다).
 23
        }
```

CardLayout 예제

```
24
25
       void addButton(String str, Container target) {
           JButton button = new JButton(str);
26
27
          button.addActionListener(this);
          target.add(button);
28
29
       }
30
       private class Cards extends JPanel {
31
          CardLayout layout;
32
33
          public Cards() {
34
              layout = new CardLayout();
35
              setLayout(layout);
36
              for (int i = 1; i <= 10; i++) {
37
                 add(new JButton("현재 카드의 번호는 " + i + "입니다"), "Center");
38
39
40
41
       }
```

CardLayout 예제

```
42
                                                                              이벤트를 처리하는 부분이다.
       public void actionPerformed(ActionEvent e) {
                                                                              사후에 학습한다.
43
          if (e.getActionCommand().equals("종료")) {
44
45
             System.exit(0);
          } else if (e.getActionCommand().equals("<<")) {</pre>
46
                                                                              CardLayout 중에서
47
             cards.layout.first(cards);
                                                                              첫 번째 카드를 선택한다.
          } else if (e.getActionCommand().equals("<")) {</pre>
48
49
             cards.layout.previous(cards);
          } else if (e.getActionCommand().equals(">")) {
50
             cards.layout.next(cards);
51
52
          } else if (e.getActionCommand().equals(">>")) {
             cards.layout.last(cards);
53
54
          }
55
56
57
58
    public class CardTest {
       public static void main(String args[]) {
59
          MyFrame f = new MyFrame();
60
61
       }
62
```





절대위치로 배치하기

■ 배치관리자를 NULL로 설정한다.

```
setlayout(null);
```

 add() 메소드를 사용하여 컴포넌트를 컨테이너 에 추가한다.

```
Button b = Button("Absolute Position Button");
add(b);
```

 setBounds() 메소드를 사용하여 절대 위치와 크 기를 지정한다.

```
b.setBounds(x, y, w, h);
```

■ 컴포넌트의 repaint() 메소드를 호출한다.

```
b.repaint();
```



```
AbsoluteTest.javat
     import java.awt.*;
 01
     import java.awt.event.*;
     import javax.swing.*;
 04
     class MyFrame extends JFrame {
 06
        JButton b1;
        private JButton b2, b3;
 07
 08
        public MyFrame() {
 09
 10
           setTitle("Absolute Position Test");
           setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 11
           setSize(300, 200);
 12
           JPanel p = new JPanel();
 13
                                                           ----- 패널의 배치 관리가를 계거한다.
           p.setLayout(null);
 14
 15
 16
           b1 = new JButton("Button #1");
           p.add(b1);
 17
           b2 = new JButton("Button #2");
 18
           p.add(b2);
 19
           b3 = new JButton("Button #3");
 20
 21
           p.add(b3);
 22
```

예제

```
23
       b1.setBounds(20, 5, 95, 30);
       24
25
       b3.setBounds(180, 15, 105, 90);
26
       add(p);
       setVisible(true);
27
28
  }
29
30
   public class AbsoluteTest {
31
     public static void main(String args[]) {
32
       MyFrame f = new MyFrame();
33
34
35 }
```



