

---

<JSTORM>

---

JAVA GUI

- 2



JSTORM  
<http://www.jstorm.pe.kr>

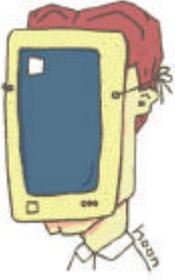
### Document Information

Document title:	JAVA GUI	- 2
Document file name:	GUI	2_ .doc
Revision number:	<1.0>	
Issued by:	< >	dwjang@jkdsoft.co.kr
	:	< > junoyoon@orgio.net
Issue Date:	<2000/8/23 >	
Status:	final	

### Content Information

Audience	GUI
Abstract	SWING - SWING SWING ..
Reference	
Benchmark information	

### Document Approvals

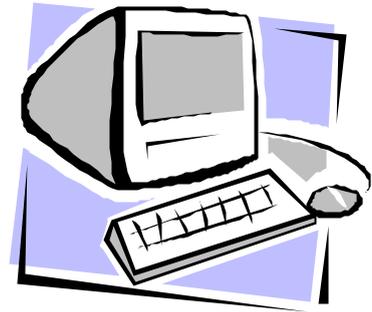
	Signature	date
		
	Signature	date

### Revision History

<u>Revision</u>	<u>Date</u>	<u>Author</u>	<u>Description of change</u>

## Table of Contents

가	.....	5
JComponent	.....	6
가 (Visibility)	(Opacity) .....	7
	.....	8
JComponent	.....	9
MVC	.....	12
Model, View, Controller	.....	13
GCad	.....	16
	.....	16
MVC	.....	17
	.....	27
(Composit)	.....	27



# GUI ! : AWT (II)

가

GUI !

AWT

GUI

AWT

가 (?)가

가

SwingSet

. ([ 1],

SwingSet

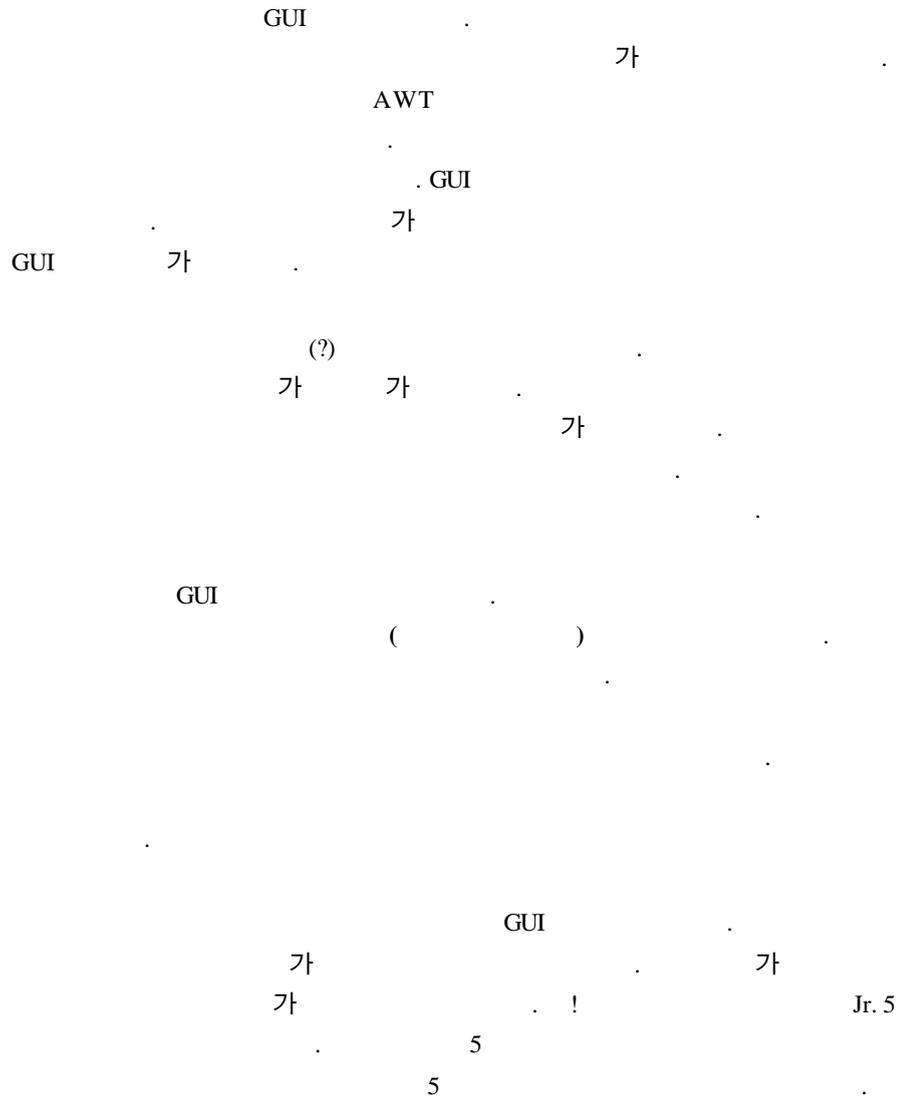
c:\jdk1.2\demo\jfc\SwingSet

.)

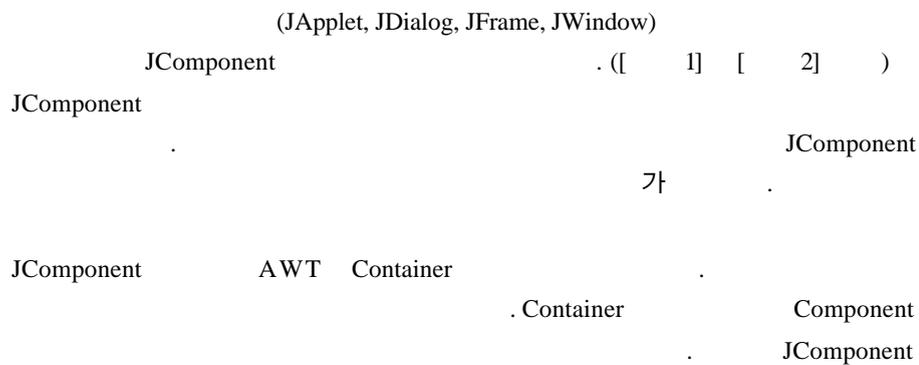
가

First Name	Last Name	Favorite Color	Favorite Sport	Favorite Fruit	Favorite Food
Mike	Albers	0, 255, 0	Soccer		44
Mark	Anthony	255, 0, 0	Baseball		2
Tom	Ball	0, 0, 255	Football		83
Alan	Chang	0, 255, 0	Baseball		838
Jeff	Dixons	255, 0, 255	Football		8
Ann	Fowler	255, 255, 0	Hockey		3
Glenn	Gerhart	0, 255, 0	Fragile		7
James	Heidberg	255, 110, 110	Tennis		21
Gari	Johnson	0, 255, 0	Bicycling		8
David	Marble	255, 0, 0	Baseball		1
David	Maths	255, 255, 0	Football		14
John	Maths	180, 180, 255	Baseball		12

[ 1]



## JComponent



Component Container  
가  
가 . JComponent 가  
.  
가 **(Visibility)** **(Opacity)**  
JComponent  
.  
.(  
99 5 319 .)  
AWT  
가  
가  
JComponent 100  
가 (Visibility) 가 가  
가  
JFrame 가 가  
frame.setVisible(true); dialog.show(); 가 가  
JButton 가  
가 button.setVisible(true);  
JButton JFrame ?  
JFrame 가 가  
가 가 가 가  
가 JButton 가  
가  
가  
AWT 가 ( 가  
)  
가 가  
setOpaque() true false 가 true  
.[ 2]  
25

( ) 가  
[ 1 ] 가  
가  
AboutDialog.java



[ 2 ]

JComponent  
가 AWT 가  
AWT 가 update() 가  
가 update() 가  
paint() update()  
update() paint() 가  
Canvas paint()  
. AWT  
AWT

```
public class GCanvas extends JPanel {  
    public void paintComponent(Graphics g) {  
        ...  
    }  
}
```



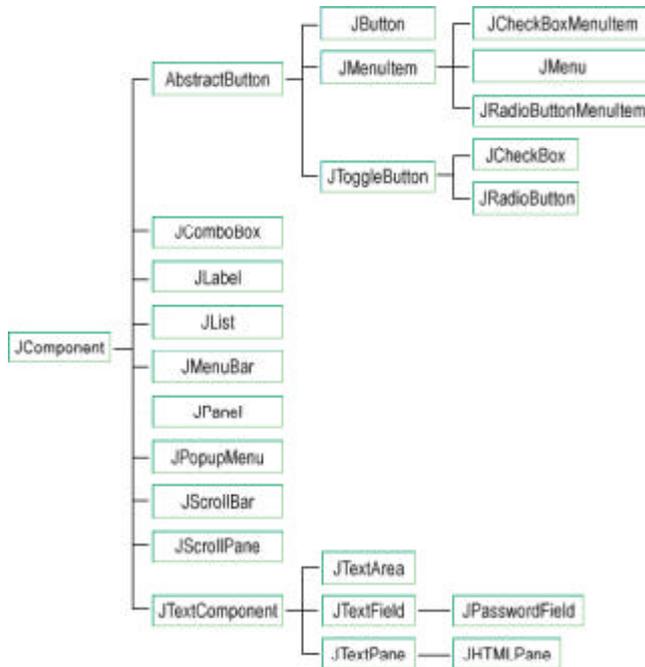
JComponent  
 JComponent  
 JComponent 가 가  
 JComponent 가 가  
 Component Container  
 [ 2] .UI  
 MVC(Model, View,  
 Controller)  
 MVC  
 ? 가  
 . [ 2]  
 true

[ 2] JComponent

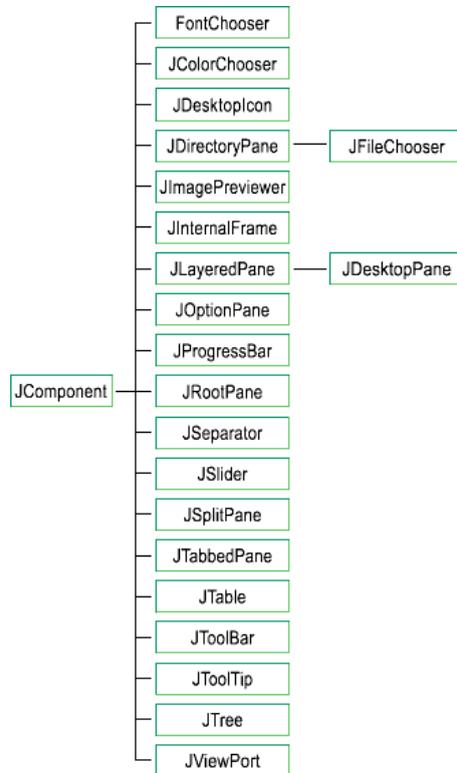
setUI()	UI (Delegate) . (MVC )
registerKeyboardAction()	
setBorder()	
setToolTipText()	
setDoubleBuffered()	

setBorder() ?

[ 1] AWT



[ 2]





[ 2] 가 ? 98 가 ?



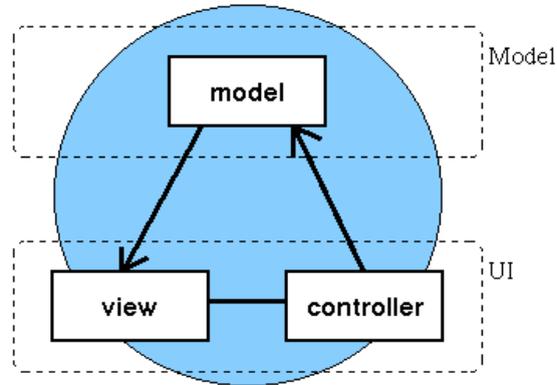
PLAF(Pluggable Look and F  
Jr. 5  
eel)

MVC ?  
(Control)  
가 ?  
가  
CIH 가

## Model, View, Controller

[ 4] 가  
가  
가 “ ” 가  
[ 4] 가 UI  
가

### JFC UI Component



[ 4] , ,

98

?

. ([ 3])

가

가

가가

가

가

가

UI (Delegate)



[ 3]

UI

Model

Model

UI

UI

JButton

ButtonModel

ButtonUI가

UI

MVC

MVC

1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

[ 4]

[ 1] ( ) .

```

public class GugudanModelDemo extends JFrame {
    public GugudanModelDemo() {
        super(" ");
        GugudanTableModel myModel = new GugudanTableModel();
        JTable table = new JTable(myModel);

        getContentPane().add(table.getTableHeader(),
            BorderLayout.NORTH);
        getContentPane().add(table, BorderLayout.CENTER);
    }
    public static void main(String[] args) {
        GugudanModelDemo frame = new GugudanModelDemo();
        frame.pack();
        frame.setVisible(true);
    }
}

//
class GugudanTableModel extends AbstractTableModel {
    <--- [2 ]
    public int getColumnCount() { return 9; }
    public int getRowCount() { return 9; }
    public String getColumnName(int col) { return "" + (col+1); }
    public Object getValueAt(int row, int col) {
        return new Integer((row+1)*(col+1));
    }
}

```

[ 1] [2 ] .



GStatus  
GStatus GMenu, GToolBar, GColorTable ?  
GStatus “ ”가  
GStatus  
~ 가  
GStatus MVC GMenu,  
GToolBar, GColorTable MVC  
가  
? GMenu, GToolBar, GColorTable GStatusBar  
GUI 가 . [ 5] . [ 7]  
가  
GStatus  
GStatus getOutlineColor()  
가  
가  
(GStatusBar) ,  
[ 5]  
“ ” 가

## MVC

. GCad, GDocument, GCanvas  
GCanvas  
GEditor ), GCanvas  
GDocument . GDocument  
GCad 가 (

```
.)  
GDocument . GDocument 가 GCanvas  
가 가 . GDocument, GCanvas, GCad  
MVC . GDocument  
GCanvas GDocument  
GCad GDocument 가  
GDocument MVC  
MVC MVC가
```

```
[ 2] GStatus  
package gcad;  
  
import gcad.gobject.*;  
import gcad.gui.*;  
import java.awt.*;  
  
////////////////////////////////////  
public class GStatus {  
    //  
    private int m_selectedShape = GShape.LINE;  
  
    //  
    private Color m_backgroundColor = Color.white;  
  
    //  
    private Color m_outlineColor = Color.black;  
  
    //  
    private int m_outlineWidth = 1;  
  
    //  
    private boolean m_isFill = false;  
  
    //  
    private Color m_fillColor = Color.white;  
  
    // Status (Controller) (View)  
    private GToolBar m_toolBar;  
    private GColorTable m_colorTable;  
    private GMenu m_menu;
```

```
private GStatusBar m_statusBar;
// -----
//
public void setToolBar(GToolBar toolBar) { m_toolBar = toolBar; }
public void setMenu(GMenu menu) { m_menu = menu; }
public void setStatusBar(GStatusBar statusBar) {
    m_statusBar = statusBar;
}
public void setColorTable(GColorTable colorTable) {
    m_colorTable = colorTable;
}
// -----
public int getSelectedShape() { return m_selectedShape; }
public void setSelectedShape(int shape) {
    m_selectedShape = shape;
    updateViews();
}
// -----
public Color getBackgroundColor() { return m_backgroundColor; }
public void setBackgroundColor(Color color) {
    m_backgroundColor = color;
    updateViews();
}
// -----
public Color getOutlineColor() { return m_outlineColor; }
public void setOutlineColor(Color color) {
    m_outlineColor = color;
    updateViews();
}
// -----
public int getOutlineWidth() { return m_outlineWidth; }
public void setOutlineWidth(int width) {
    m_outlineWidth = width;
}
// -----
public boolean isFill() { return m_isFill; }
public void setFill(boolean fill) {
    m_isFill = fill;
    updateViews();
}
// -----
```

```
public Color getFillColor() { return m_fillColor; }
public void setFillColor(Color color) {
    m_fillColor = color;
    updateViews();
}
// -----
//
private void updateViews() {
    m_toolBar.update();
    m_colorTable.update();
    m_menu.update();
    m_statusBar.update();
}
}
```

```
[ 2] [1 ] ? GStatus
가 가
.[1 ]
[ 2 ] update()
[ 3] [1 ] update()
GStatus 가
MVC가
?
[ 3] [2 ] ? 가
가 (Static initializer)
(Attribute) 가
?
. {}
```

```
[ 3] GColorTable
package gcad.gui;

import gcad.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
```

```
////////////////////////////////////  
public class GColorTable extends JPanel {  
  
    /**      (      가      ,      ) */  
    private static final Dimension TABLE_SIZE = new Dimension(2, 8);  
    /**      */  
    private static final Dimension CELL_SIZE = new Dimension(30, 30);  
    /**      */  
    private static final Color[] CELL_COLORS = {  
        new Color( 0, 0 , 0),  
        new Color(128, 0 , 0),  
        new Color( 0, 128, 0),  
        new Color(128, 128, 0),  
        new Color( 0, 0, 128),  
        new Color(128, 0, 128),  
        new Color( 0, 128, 128),  
        new Color(192, 192, 192),  
        new Color(128, 128, 128),  
        new Color(255, 0, 0),  
        new Color( 0, 255, 0),  
        new Color(255, 255, 0),  
        new Color( 0, 0, 255),  
        new Color(255, 0, 255),  
        new Color( 0, 255, 255),  
        new Color(255, 255, 255)  
    };  
  
    /*  
    private static final Color[] CELL_COLORS =  
        new Color[TABLE_SIZE.height*TABLE_SIZE.width];  
  
    // Static initializer      <<--- [2 ]  
    {  
        int colorCount = TABLE_SIZE.height*TABLE_SIZE.width;  
        for (int i=0; i<colorCount; i++) {  
            CELL_COLORS[i] = new Color(i*(255/colorCount), 0, 0);  
        }  
    }  
    */  
    // Status
```

```
private GStatus m_status;

/**          */
private SelectedColorCell m_selectedColorCell;

// -----
public GColorTable(GStatus status) {
    m_status = status;
    m_status.setColorTable(this);
    setBorder(BorderFactory.createEtchedBorder());

    //
    JPanel standardColorsPane = new JPanel();
    standardColorsPane.setLayout(new GridLayout(0,
        TABLE_SIZE.width));
    for (int i=0; i<CELL_COLORS.length; i++) {
        standardColorsPane.add(new ColorCell(CELL_COLORS[i]));
    }

    //
    m_selectedColorCell = new SelectedColorCell();
    m_selectedColorCell.addMouseListener(new MouseAdapter() {
        public void mouseClicked(MouseEvent e) {
            Color oldColor = m_status.getOutlineColor();
            Color newColor;
            newColor = JColorChooser.showDialog(null,
                "          ", oldColor);
            m_status.setOutlineColor(newColor);
        }
    });

    //
    JPanel colorPane = new JPanel();
    colorPane.setLayout(new BorderLayout());
    colorPane.add(standardColorsPane, BorderLayout.NORTH);
    colorPane.add(new JSeparator(SwingConstants.VERTICAL));
    colorPane.add(m_selectedColorCell, BorderLayout.SOUTH);
    setLayout(new BorderLayout());
    add(colorPane, BorderLayout.NORTH);
}
```

```
// -----  
public void update() {                                     <!-- [1  ]  
    Color color = m_status.getOutlineColor();  
    m_selectedColorCell.repaint();  
}  
  
////////////////////////////////////  
/**                                     */  
private class ColorCell extends JPanel {  
  
    /**                                     */  
    private Color m_color;  
  
    // -----  
    //      :  
    public ColorCell(Color color) {  
        m_color = color;  
        setBorder(BorderFactory.createEtchedBorder());  
        setPreferredSize(CELL_SIZE);  
  
        addMouseListener(new MouseAdapter() {  
            public void mouseClicked(MouseEvent e) {  
                m_status.setOutlineColor(m_color);  
            }  
        });  
    }  
  
    // -----  
    public void paintComponent(Graphics g) {  
        g.setColor(m_color);  
        g.fillOval(2, 2, getSize().width-4, getSize().height-4);  
    }  
}  
////////////////////////////////////  
private class SelectedColorCell extends JPanel {  
  
    // -----  
    public SelectedColorCell() {  
        setBorder(BorderFactory.createEtchedBorder());  
        setPreferredSize(new Dimension(  
            2*CELL_SIZE.width, 2*CELL_SIZE.height));  
    }  
}
```

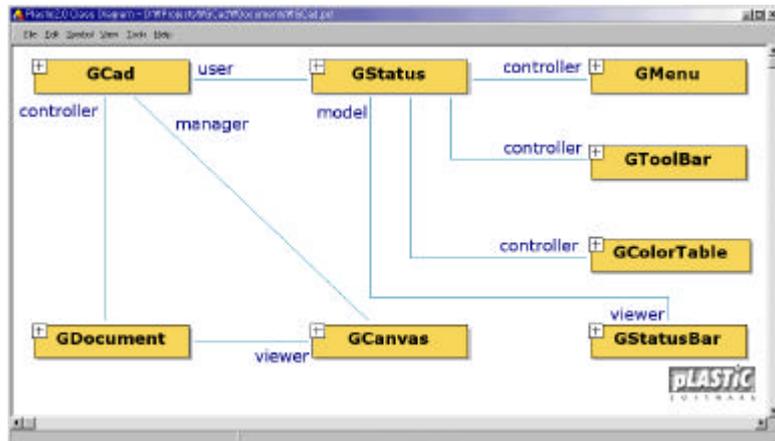
```
    }  
  
    // -----  
    public void paintComponent(Graphics g) {  
        Color color = m_status.getOutlineColor();  
        g.setColor(color);  
        g.fillRect(2, 2, getSize().width-4, getSize().height-4);  
    }  
}  
}
```

#### [ 4] GDocument

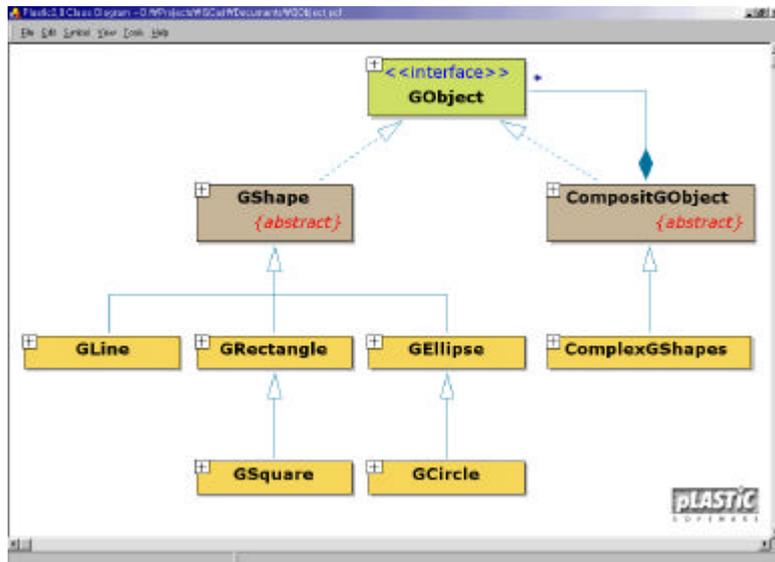
```
package gcad;  
  
import gcad.gobject.*;  
import java.util.*;  
import java.io.*;  
  
////////////////////////////////////  
public class GDocument {  
  
    //  
    private Vector m_shapes = new Vector();  
  
    // -----  
    //      가  
    public void addShape(GShape shape) { m_shapes.addElement(shape); }  
  
    // -----  
    //  
    public Enumeration getShapes() { return m_shapes.elements(); }  
  
    // -----  
    //  
    public void clear() { m_shapes.removeAllElements(); }  
  
    // -----  
    //  
    public void save() {}  
  
    // -----  
}
```

```
//  
public void load() {}  
}
```

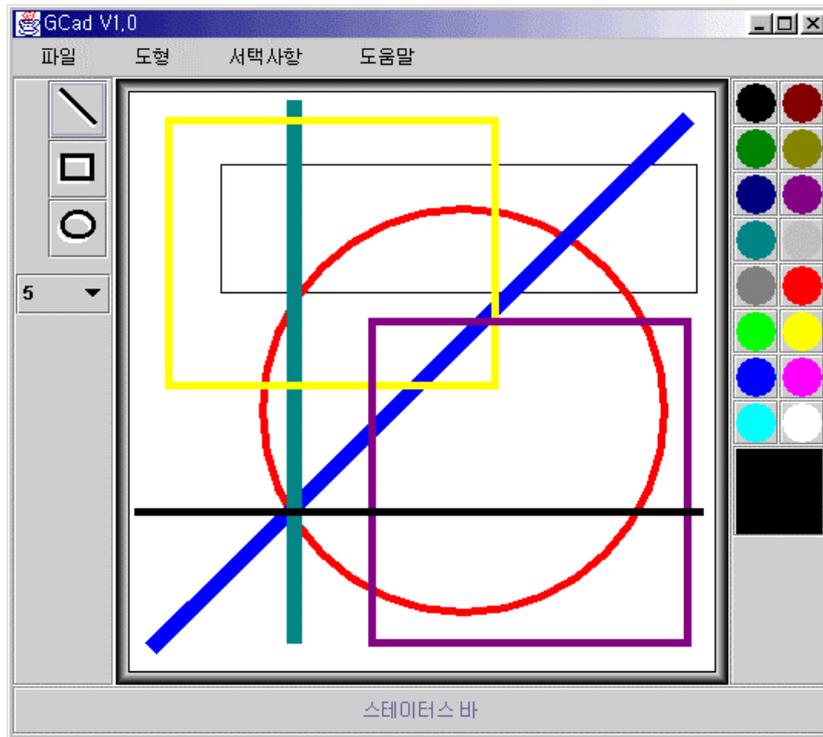
[ 5 ] GCad



[ 6 ]

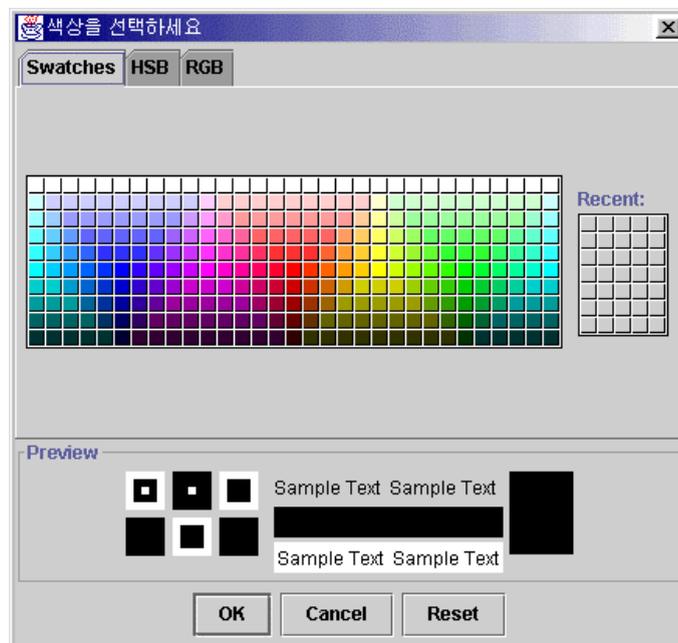


[ 7 ] GCad



GCad

~~~



가 . [ 8]  
가 가

가 CompositGObject

### (Composit)

CompositGObject

GCad

(Grouping)

가

GCrossedRectangle

가

?

가

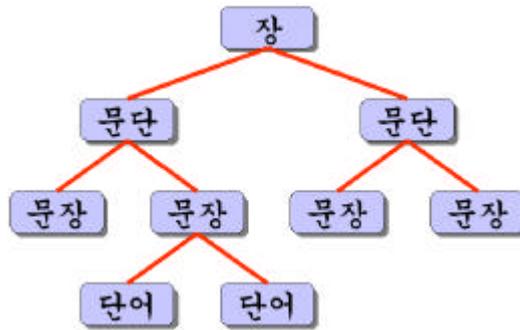
(JTextField, JTextArea, JEditorPane

JTextPane ) [ 5]

(Composit)

3 260

[ 5]



[ 6] CompositGObject 가 GObject  
 (Composit) CompositGObject GObject  
 (implements) CompositGObject GObject  
 (is a) GObject

[ 5] [ 6] UML ( )  
 . UML http://www.rational.com/uml/  
 12 374  
 (implements) (extends)  
 (Composit Aggregation)

GLine . GObject, GShape,

[ 7] [ 1 ] .2D  
 2D GEditor Graphics가  
 2D  
 가 가 JDK 가  
 2D 가

[ 5] GObject

```
package gcad.gobject;

import java.awt.*;
import java.io.*;

////////////////////////////////////
public interface GObject extends Serializable {

    // -----
    //
    public abstract void draw(Graphics g);

    // -----
    //
    public abstract void erase(Graphics g);
}

```

```
[ 6] GShape
package gcad.gobject;

import java.awt.*;

////////////////////////////////////
public abstract class GShape implements GObject {

    //
    public static final int LINE    = 1;
    public static final int RECTANGLE = 2;
    public static final int SQUARE  = 3;
    public static final int ELLIPSE  = 4;
    public static final int CIRCLE   = 5;

    //
    protected Color m_outlineColor = Color.black;

    //
    protected int m_outlineWidth = 1;

    //
    protected boolean m_isFill = false;
}

```

```
//
protected Color m_fillColor = Color.white;

// -----
//
public Color getOutlineColor() { return m_outlineColor; }
public void setOutlineColor(Color color) { m_outlineColor = color; }
public int getOutlineWidth() { return m_outlineWidth; }
public void setOutlineWidth(int width) { m_outlineWidth = width; }
public boolean isFill() { return m_isFill; }
public void setFill(boolean fill) { m_isFill = fill; }
public Color getFillColor() { return m_fillColor; }
public void setFillColor(Color color) { m_fillColor = color; }

// -----
// ( )
public abstract void setPenPosition(Point point);

// -----
//
public void erase(Graphics g) {
    g.setXORMode(Color.white);
    draw(g);
}
}

[ 7] GLine
package gcad.gobject;

import java.awt.*;

////////////////////////////////////
public class GLine extends GShape {

//
private Point m_startPoint;

//
private Point m_endPoint;

// -----
```

```
public GLine(Point point) {
    m_startPoint = point;
    m_endPoint = point;
}

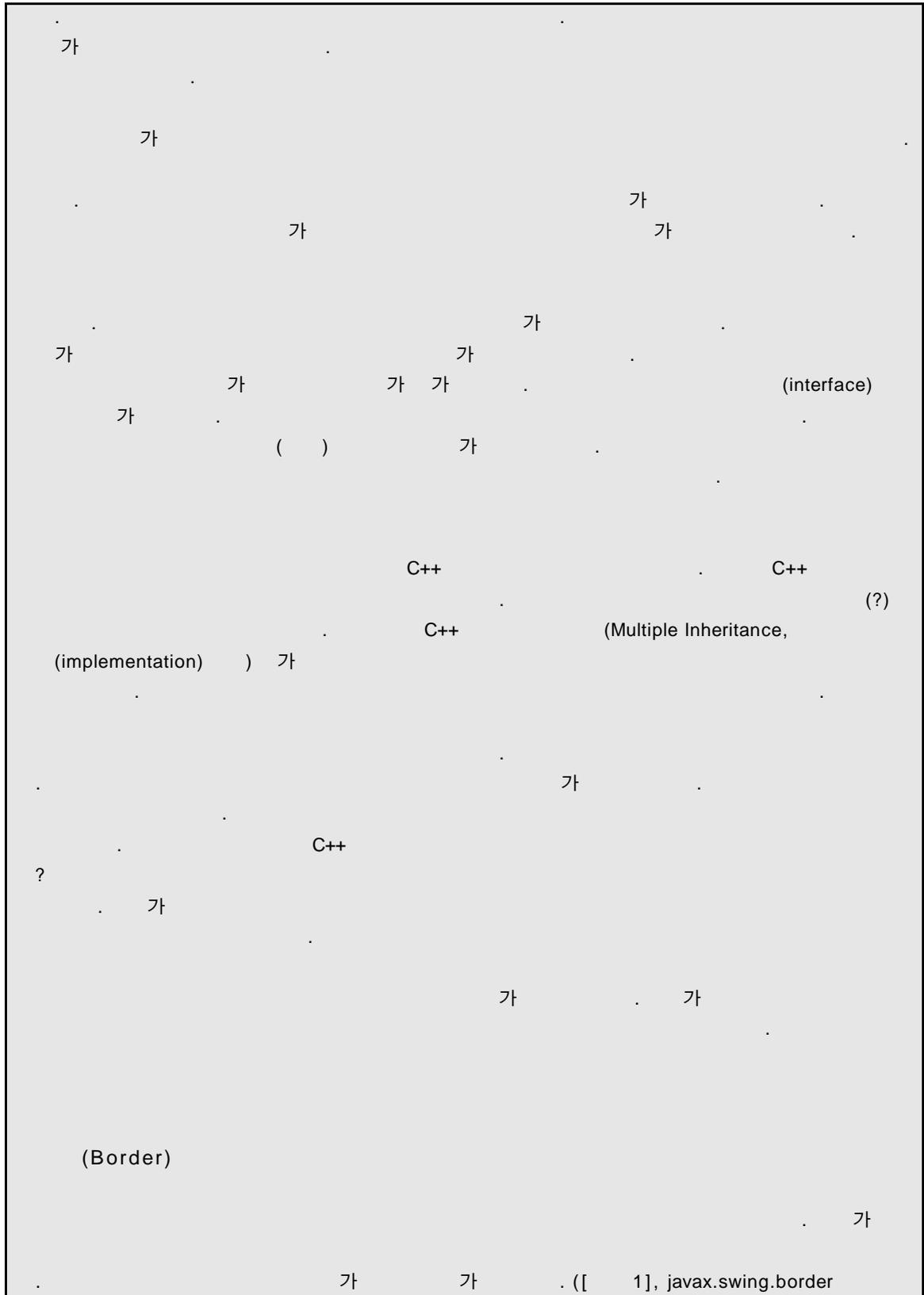
// -----
// ( )
public void setPenPosition(Point point) {
    m_endPoint = point;
}

// -----
//
public void draw(Graphics g) {
    Graphics2D g2 = (Graphics2D)g;           <---- [1 ]

    Stroke oldStroke = g2.getStroke();
    g2.setStroke(new BasicStroke(m_outlineWidth));
    g2.setColor(m_outlineColor);
    g2.drawLine(m_startPoint.x, m_startPoint.y,
                m_endPoint.x, m_endPoint.y);
    g2.setStroke(oldStroke);
}
}

GCad
```





?)

?

[ 1 ]

a) javax.swing.border import

b) AbstractBorder

c)

```
public void paintBorder(Component c, Graphics g, int x, int y, int dx, int dy)
public Insets getBorderInsets(Component c)
```

d) setBorder

[ 2 ] [ 1 ] 가

가 (GrayGradientBorder) 가

. paintBorder() 가

x y . pai

ntBorder() 가 JComponent paintBo

rder() . ( [ 1 ] ) JComponent p

aint() . paint() paintBorder()

가

[ 1 ]

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.*;
```

```
////////////////////////////////////  
public class GrayGradientBorder extends AbstractBorder {  
  
    //  
    private int m_width = 10;  
  
    // -----  
    // .  
    //  
    // c -      가  
    // g -      Graphics  
    // x, y -  
    // dx, dy -  
    public void paintBorder(Component c, Graphics g,  
        int x, int y, int dx, int dy) {  
  
        Color[] colors = new Color[m_width];  
        colors[0] = new Color(255, 255, 255);  
        colors[colors.length - 1] = new Color(0, 0, 0);  
  
        float density;  
        for (int i=1; i<colors.length - 1; i++) {  
            density = (float)(i - 1) / (float)(m_width - 2);  
            colors[i] = new Color(density, density, density);  
        }  
  
        for (int i=0; i<colors.length; i++) {  
            g.setColor(colors[i]);  
            g.drawRect(x+i, y+i, dx - i*2 - 1, dy - i*2 - 1);  
        }  
    }  
  
    // -----  
    // .  
    // 가  
  
    public Insets getBorderInsets(Component c) {  
        return new Insets(  
            m_width, m_width, m_width, m_width);  
    }  
  
    // -----  
    public static void main(String[] args) {
```

```
JFrame frame = new JFrame("Border Test");
JButton button = new JButton("Bordered Button");

button.setBackground(Color.cyan);
button.setBorder(new GrayGradientBorder());
frame.getContentPane().setLayout(new BorderLayout());
frame.getContentPane().add(button, BorderLayout.CENTER);

frame.setSize(300, 200);
frame.setVisible(true);
}
}
```

[ 2 ]



...

가 가 가

3

Test.java:12: Variable url may not have been initialized.  
System.out.println("This is url " + url);

(Exception Handling)

가

?

가

가

try

catch

```
import java.net.*;
public class Test {

    public static void main(String[] args) {
        URL url;
        try {
            url = new URL("http://www.cse.cau.ac.kr/");
        } catch (MalformedURLException e) {
            System.out.println("Bad URL:" + e.getMessage());
        }
        System.out.println("This is url " + url);
    }
}
```

[1 ] url 가 null 가

가

[2 ] url 가

? 가 ?

URL URL MalformedURLException (throw)

try 가 try

try [2 ]

가 url

null [1 ]

URL url = null;

|     |   |   |   |
|-----|---|---|---|
| url | 가 | . | 가 |
|-----|---|---|---|