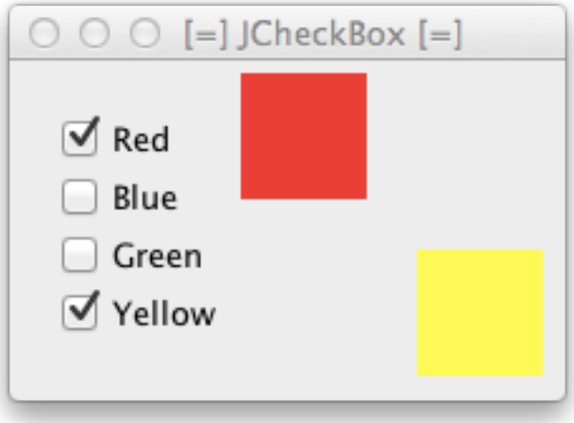
Graphical User Interface using Swing
Part III - More Swing Widgets

Samir Genaim

http://docs.oracle.com/javase/tutorial/uiswing/
http://www.macs.hw.ac.uk/guidebook

JCheckBox

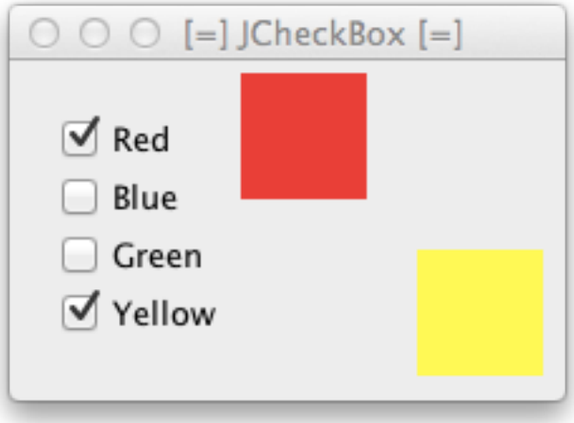
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

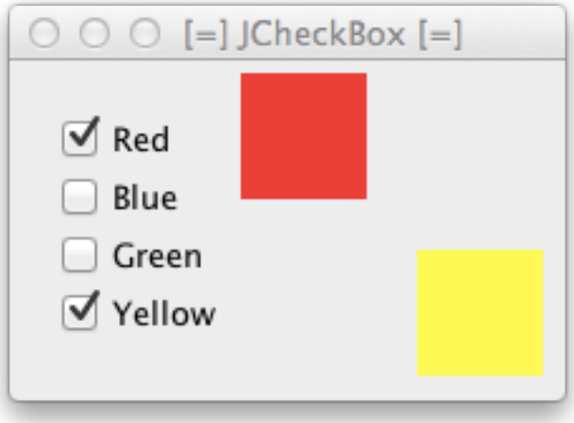
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

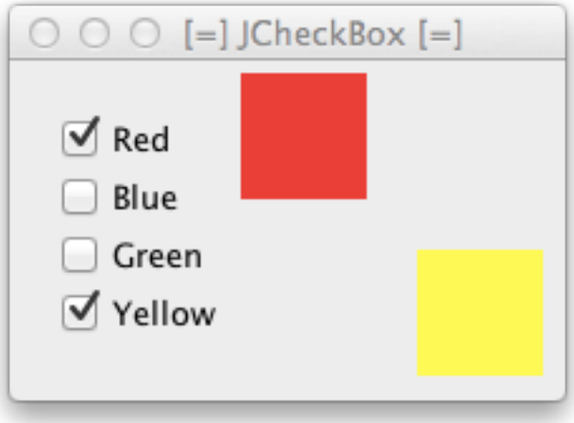
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

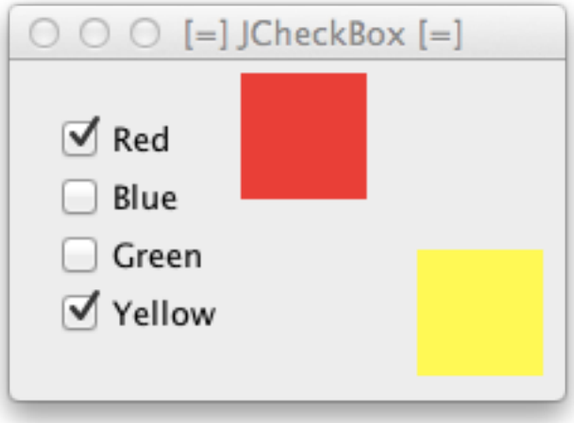
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

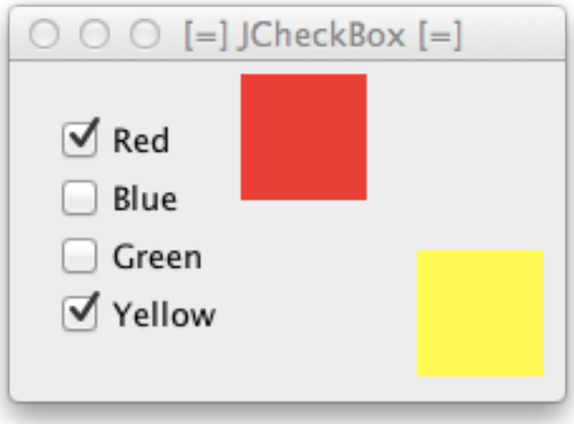
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

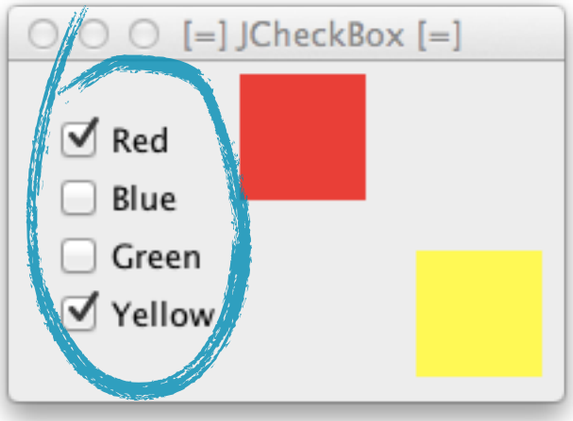
```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



see: examples.swing.misc.CheckBoxExample

JCheckBox

```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```

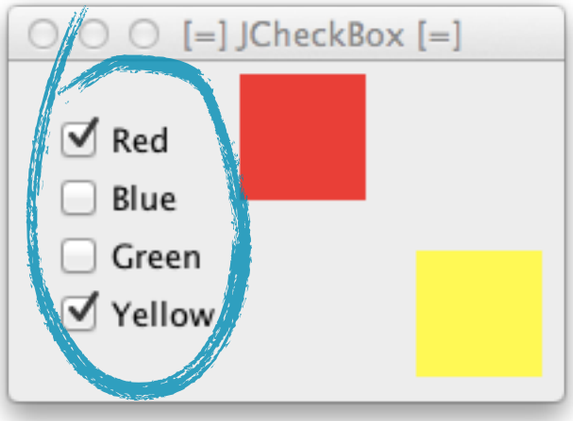


Use **BoxLayout** to group checkboxes

see: examples.swing.misc.CheckBoxExample

JCheckBox

```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



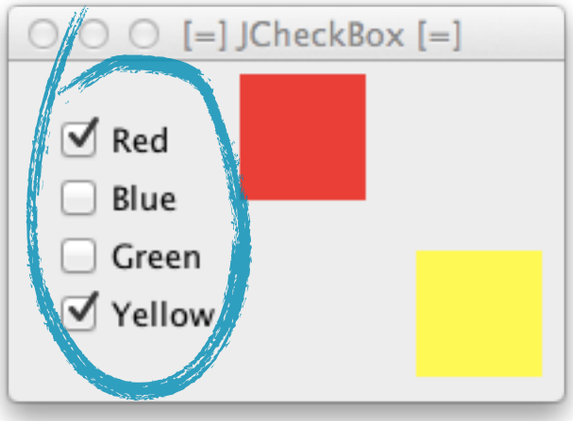
Use **BoxLayout** to group checkboxes

Why we use ItemListener and not ActionListener?

see: examples.swing.misc.CheckBoxExample

JCheckBox

```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;

        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



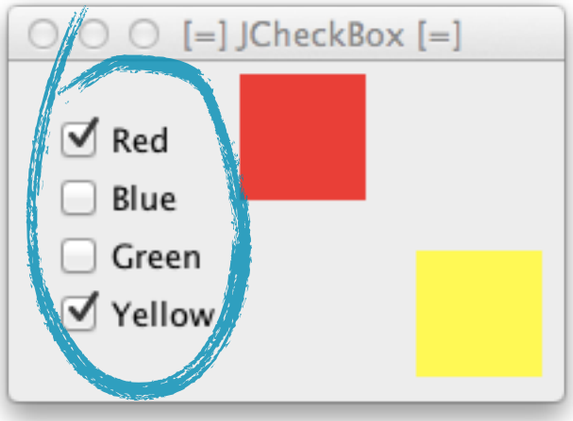
Use **BoxLayout** to group checkboxes

Why we use ItemListener and not ActionListener?

see: examples.swing.misc.CheckBoxExample

JCheckBox

```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



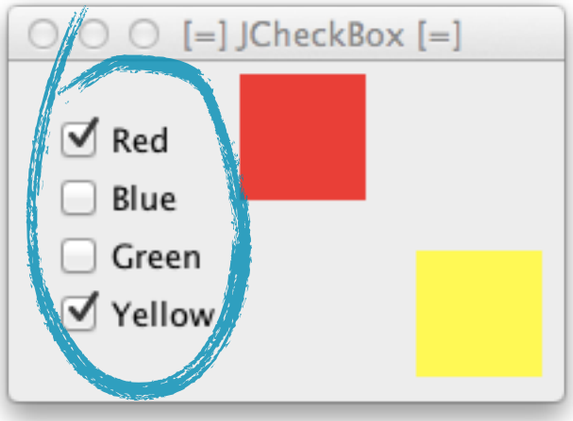
Use **BoxLayout** to group checkboxes

Why we use ItemListener and not ActionListener?

see: examples.swing.misc.CheckBoxExample

JCheckBox

```
public class CheckBoxExample extends JFrame implements ItemListener {

    JCheckBox redCB, blueCB, greenCB, yellowCB;
    JPanel redBox, blueBox, greenBox, yellowBox;
    ...
    private void initGUI() {
      ...
        redCB = new JCheckBox("Red");
        redCB.setSelected(true);
        redCB.addItemListener(this);
      ...
        blueCB = new JCheckBox("Blue");
        blueCB.setSelected(false);
        blueCB.addItemListener(this);
        ...
    }

    public void itemStateChanged(ItemEvent e) {
        boolean visible = false;
        if (e.getStateChange() == ItemEvent.SELECTED) visible = true;

        if(e.getItemSelectable() == redCB) redBox.setVisible(visible);
        else if(e.getItemSelectable() == blueCB) blueBox.setVisible(visible);
        else if(e.getItemSelectable() == greenCB) greenBox.setVisible(visible);
        else if(e.getItemSelectable() == yellowCB) yellowBox.setVisible(visible);
    }
}
```



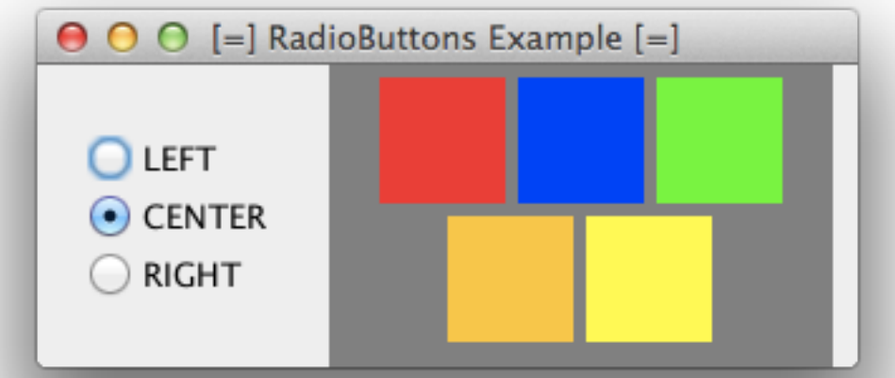
Use **BoxLayout** to group checkboxes

Why we use ItemListener and not ActionListener?

see: examples.swing.misc.CheckBoxExample

JRadioButton

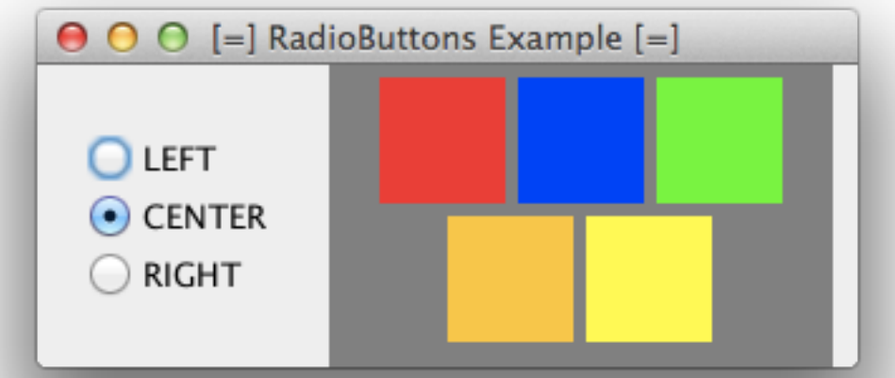
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  …
    private void initGUI() {
        …
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) …
        else if ( e.getSource() == centerButton ) …
        else if ( e.getSource() == rightButton ) …
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

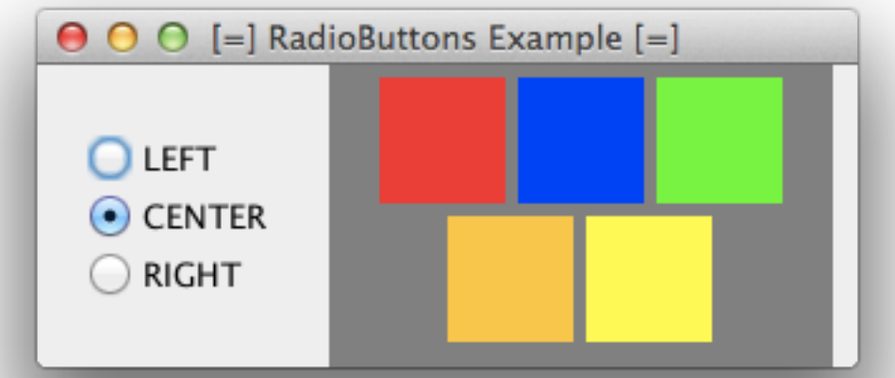
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  …
    private void initGUI() {
        …
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) …
        else if ( e.getSource() == centerButton ) …
        else if ( e.getSource() == rightButton ) …
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

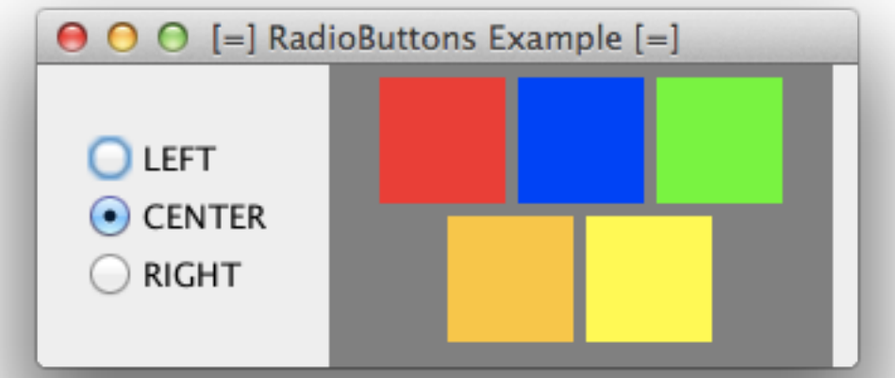
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  …
    private void initGUI() {
        …
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) …
        else if ( e.getSource() == centerButton ) …
        else if ( e.getSource() == rightButton ) …
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

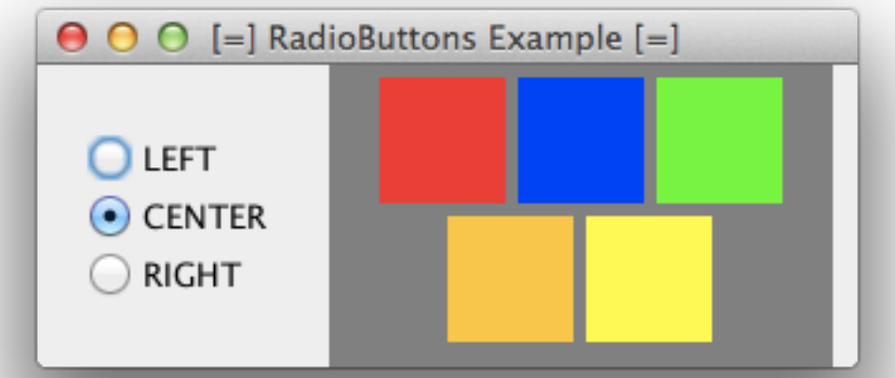
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  …
    private void initGUI() {
        …
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) …
        else if ( e.getSource() == centerButton ) …
        else if ( e.getSource() == rightButton ) …
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

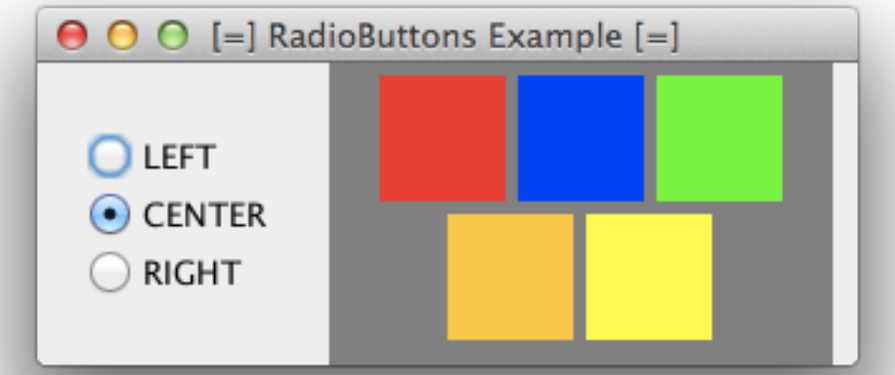
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  …
    private void initGUI() {
        …
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) …
        else if ( e.getSource() == centerButton ) …
        else if ( e.getSource() == rightButton ) …
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

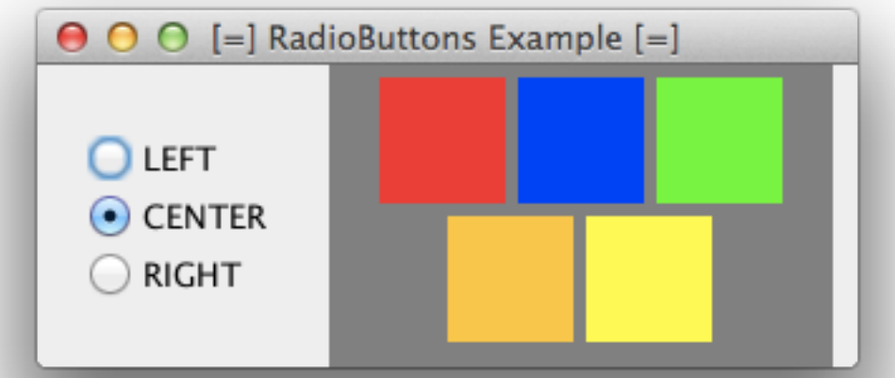
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  ...
    private void initGUI() {
        ...
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) ...
        else if ( e.getSource() == centerButton ) ...
        else if ( e.getSource() == rightButton ) ...
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

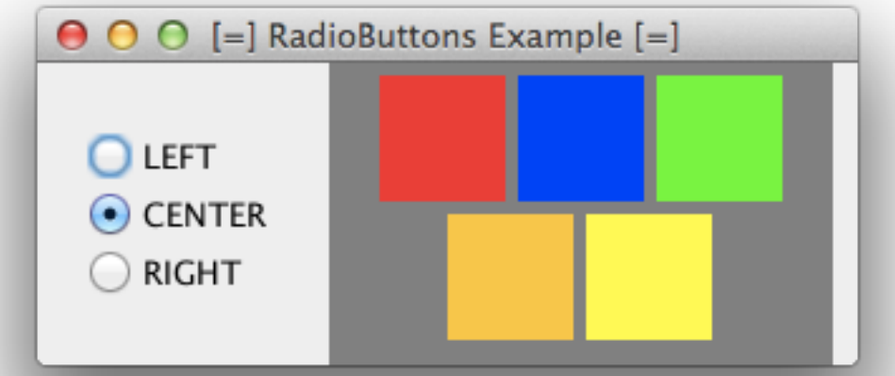
```
public classleft RadioButtonExample extends JFrame implements  ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  ...
    private void initGUI() {
        ...
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) ...
        else if ( e.getSource() == centerButton ) ...
        else if ( e.getSource() == rightButton ) ...
    }
}
```



see: examples.swing.misc.RadioButtonExample

JRadioButton

```
public classleft RadioButtonExample extends JFrame implements ActionListener {

    JRadioButton leftButton, centerButton, rightButton;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
  ...
    private void initGUI() {
        ...
        leftButton = new JRadioButton(names[0]);
        leftButton.addActionListener(this);
        leftButton.setSelected(true);
        centerButton = new JRadioButton(names[1]);
        centerButton.addActionListener(this);
        rightButton = new JRadioButton(names[2]);
        rightButton.addActionListener(this);

        ButtonGroup btnGroup = new ButtonGroup();
        btnGroup.add(leftButton);
        btnGroup.add(centerButton);
        btnGroup.add(rightButton);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == leftButton ) ...
        else if ( e.getSource() == centerButton ) ...
        else if ( e.getSource() == rightButton ) ...
    }
}
```

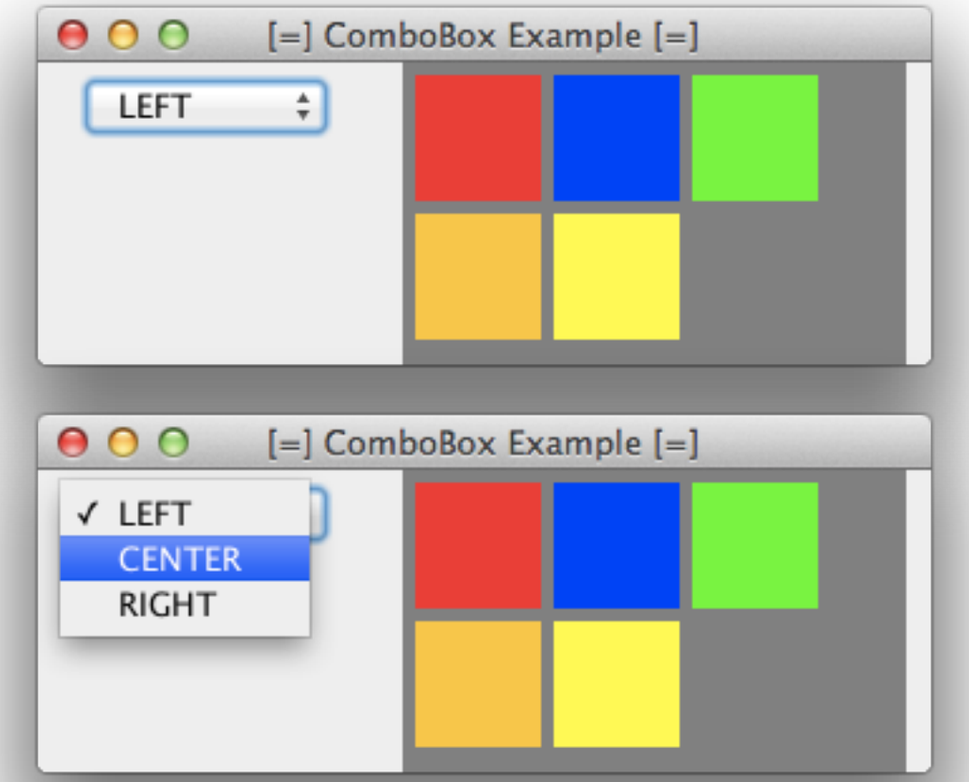


What is the difference between **ActionListener** and **ItemListener** in this case? How many times each is called when selecting a different option?

see: examples.swing.misc.RadioButtonExample

JComboBox

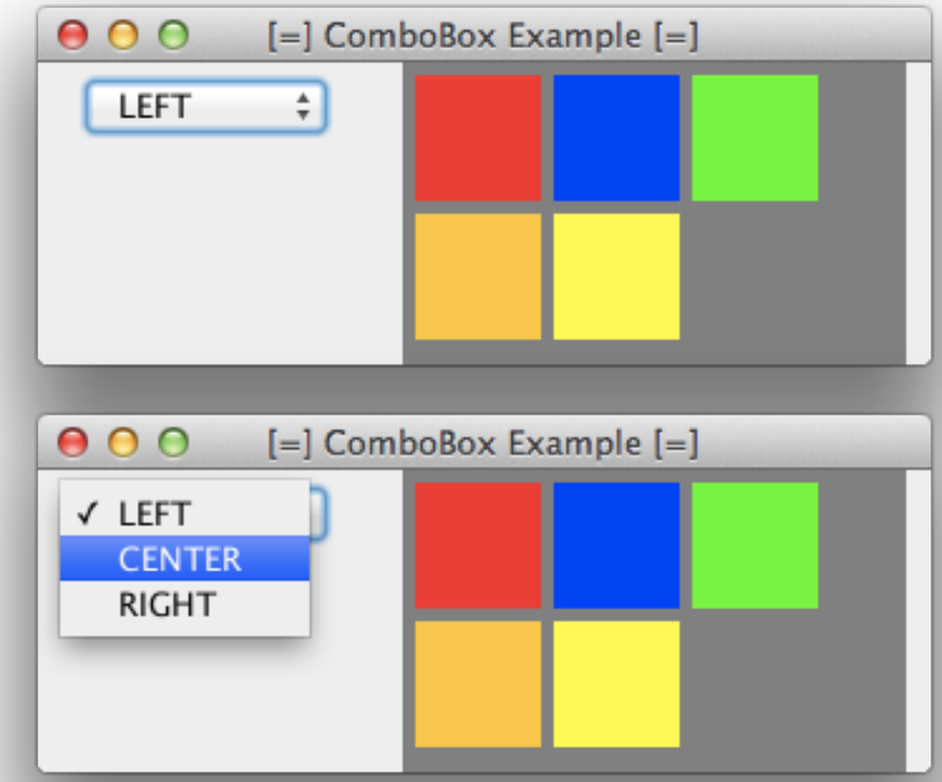
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

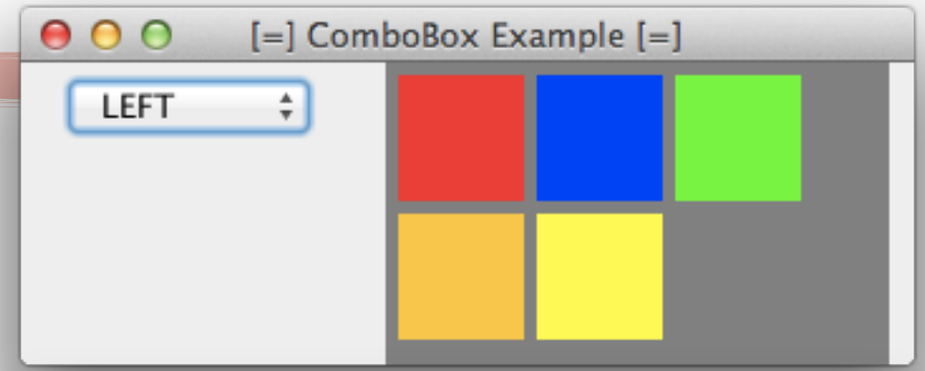
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

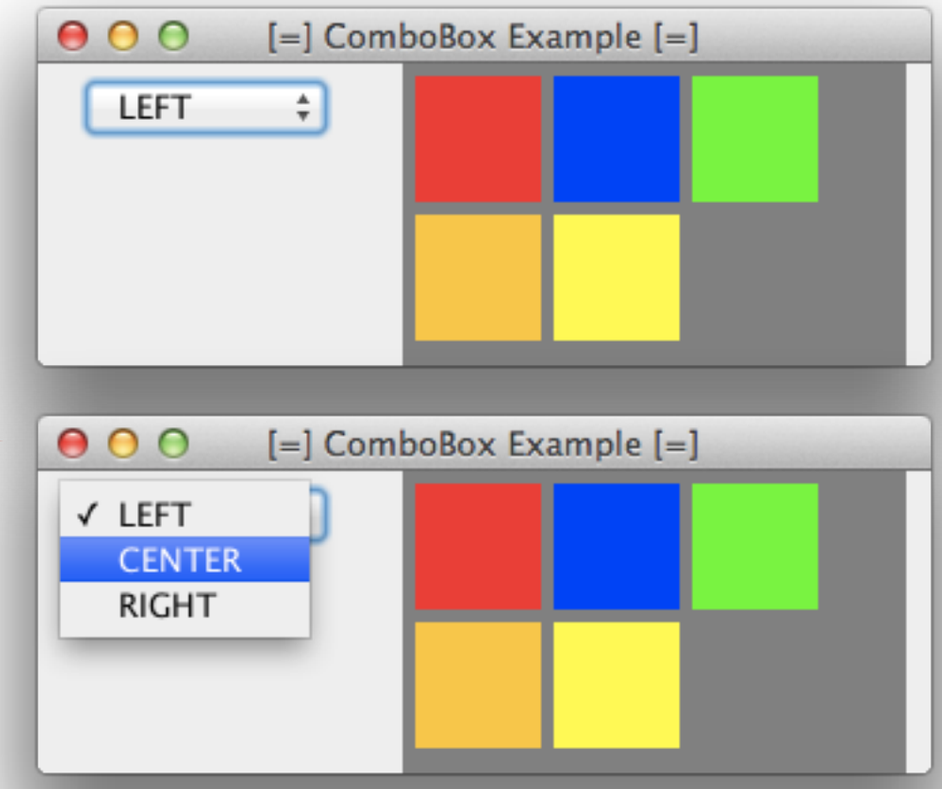
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    ...

    private void initGUI () {
        ...
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        ...
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  ...
    else if ( name.equals( names[1] ) ) ...
        else if ( name.equals( names[2] ) ) ...
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

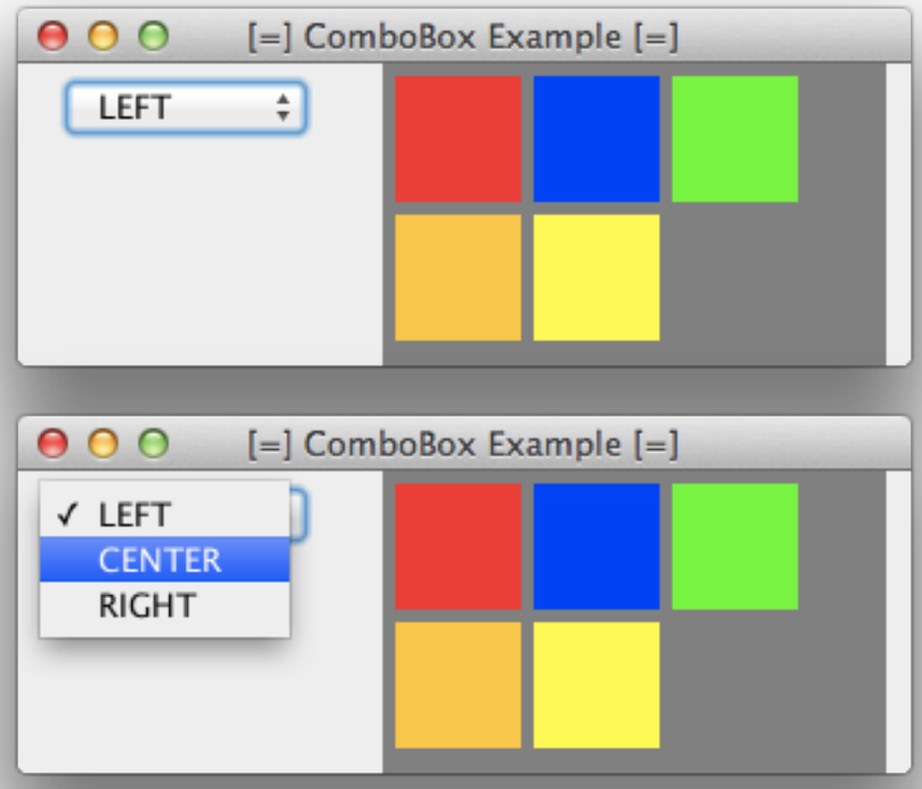
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

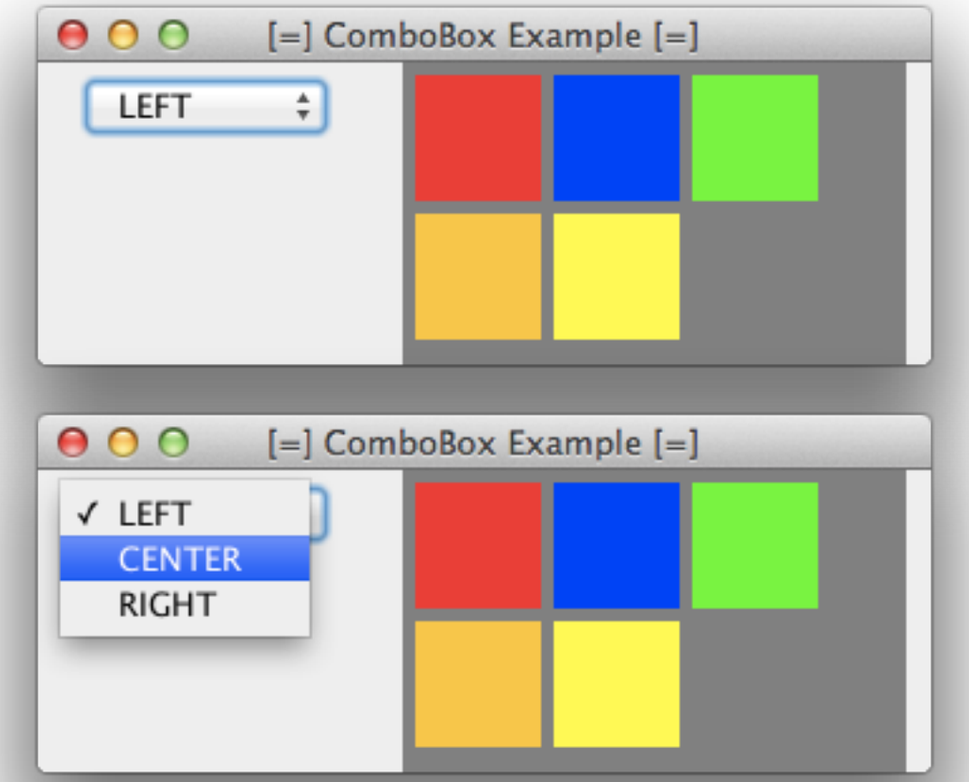
JComboBox

```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```

see: examples.swing.misc.ComboBoxExample

JComboBox

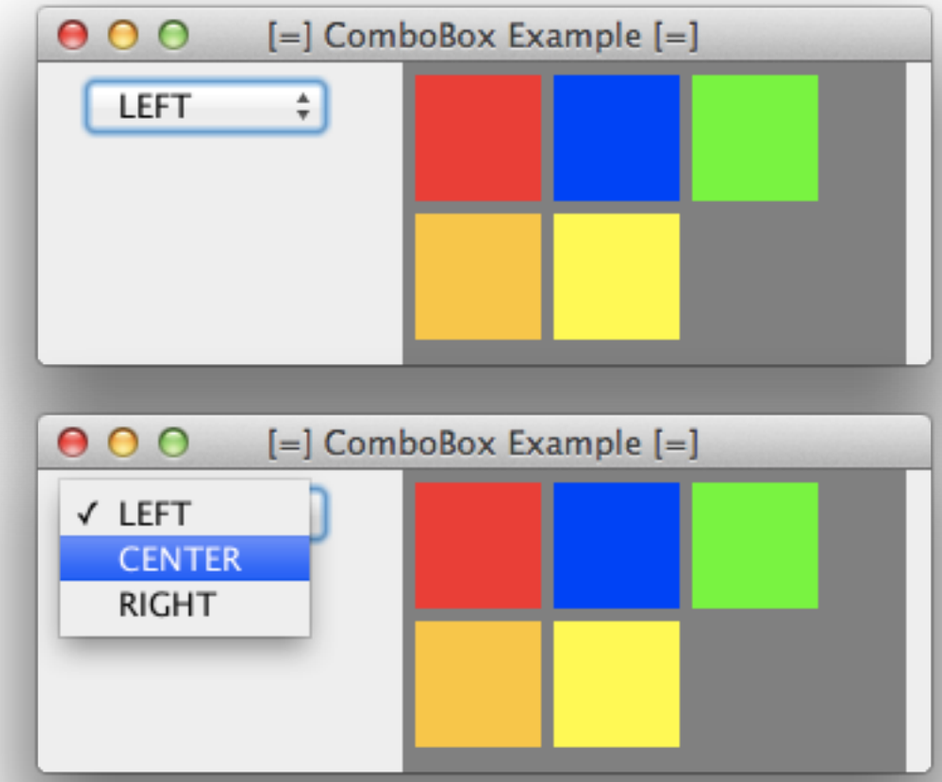
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

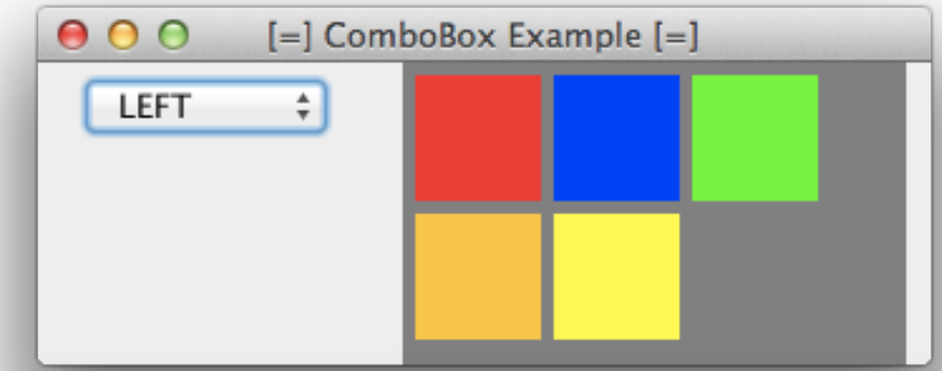
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

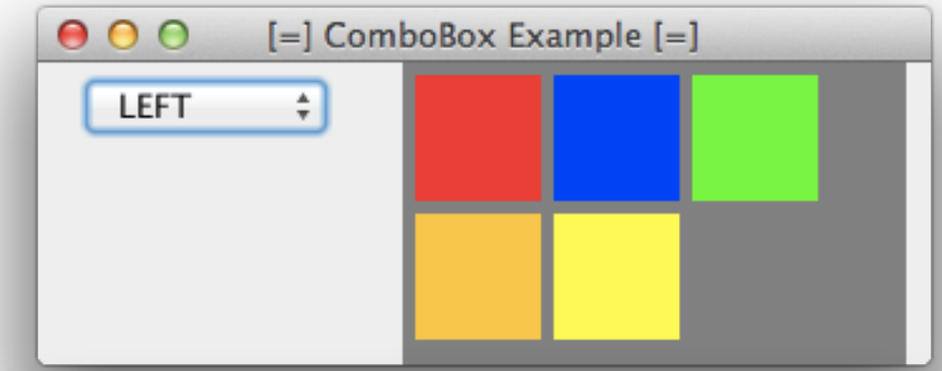
```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



see: examples.swing.misc.ComboBoxExample

JComboBox

```
public class ComboBoxExample extends JFrame implements  ActionListener {

    JComboBox<String> list;
    String names[] = {"LEFT", "CENTER", "RIGHT"};
    …

    private void initGUI () {
        …
        JComboBox list = new JComboBox(names);
        list.setSelectedIndex(0);
        list.addActionListener(this);
        …
    }

    public void actionPerformed(ActionEvent e) {

        JComboBox<String> cb = (JComboBox<String>)e.getSource();
        String name = (String)cb.getSelectedItem();

        if ( name.equals( names[0] ) )  …
    else if ( name.equals( names[1] ) ) …
        else if ( name.equals( names[2] ) ) …
    }
}
```



String can be changed to any Object, toString will be used to display the names of the items

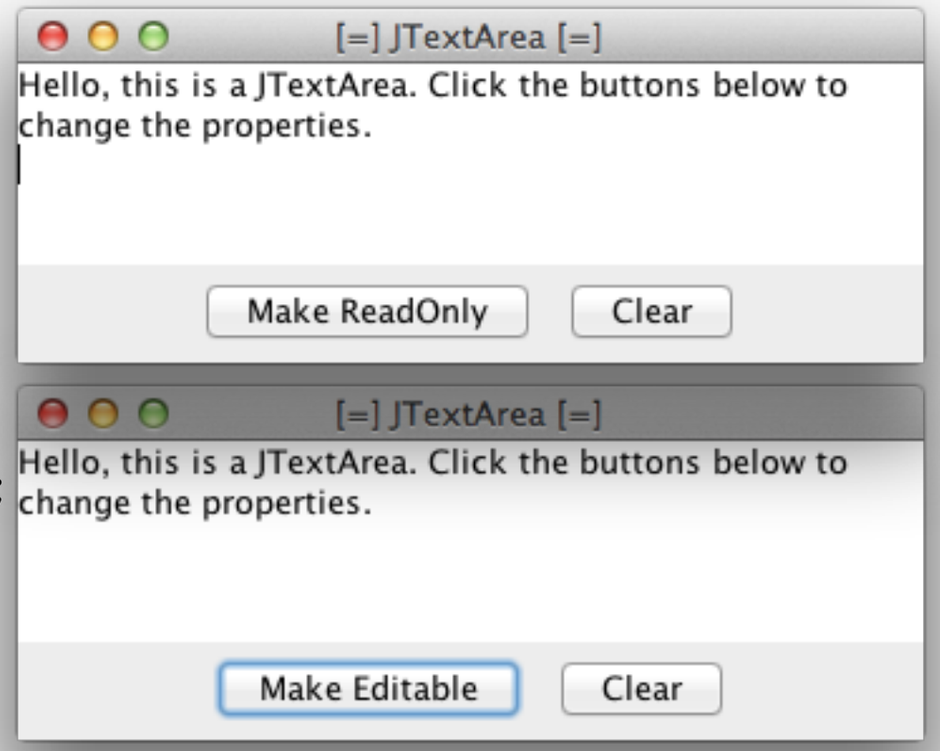
Add/remove items dynamically

```
list.addItem(…)
list.removeItem(…)
```

see: examples.swing.misc.ComboBoxExample

JTextArea

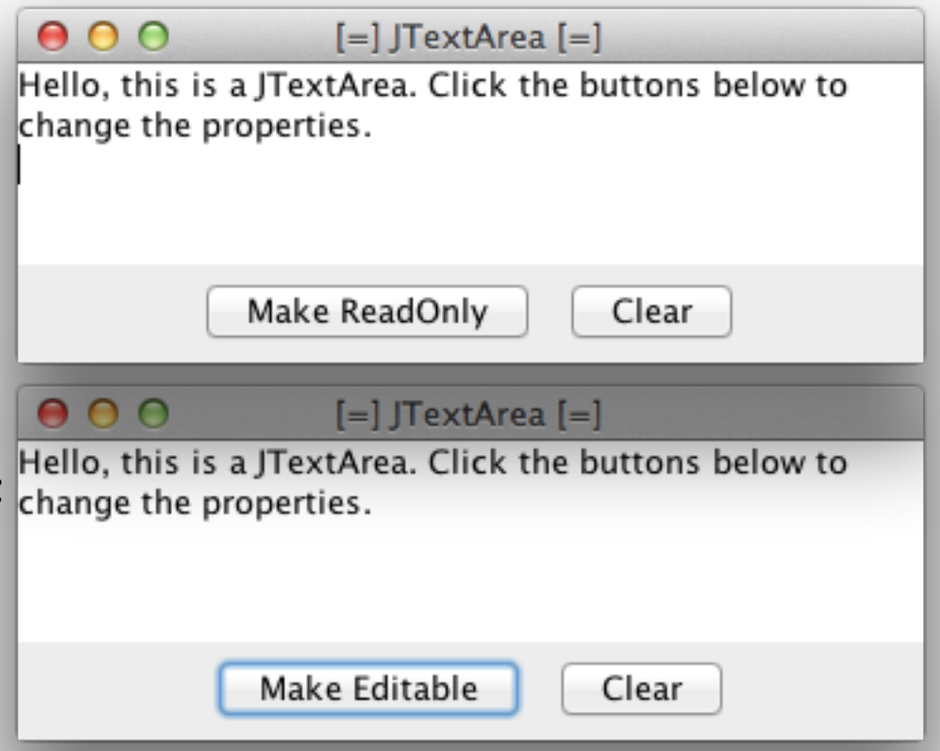
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  …
    private void initGUI() {
      …
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

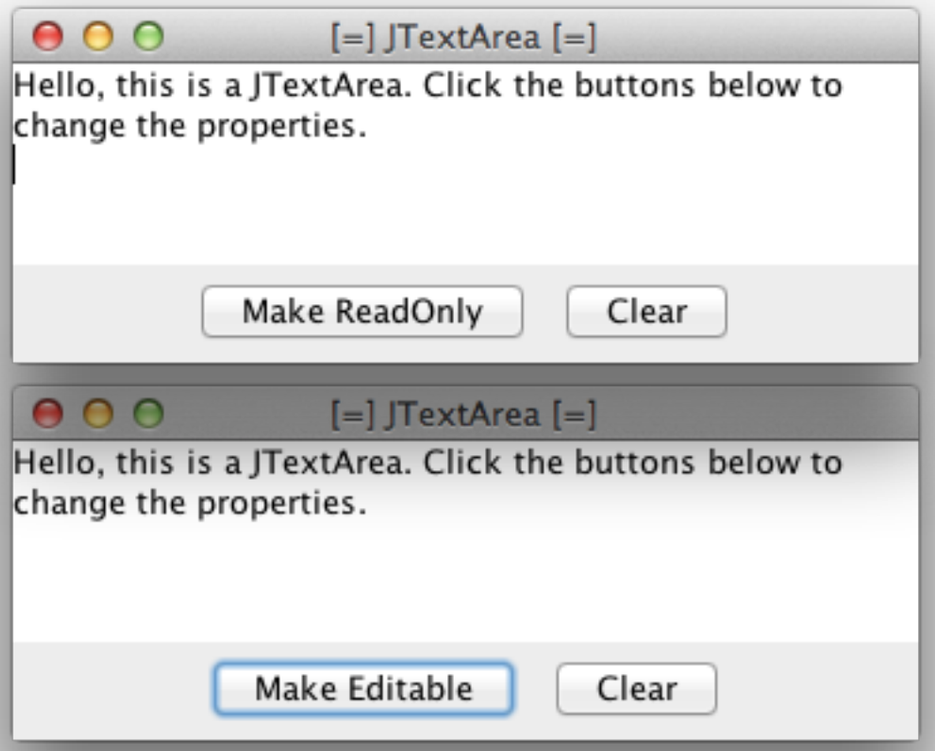
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
    ...
    private void initGUI() {
        ...
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

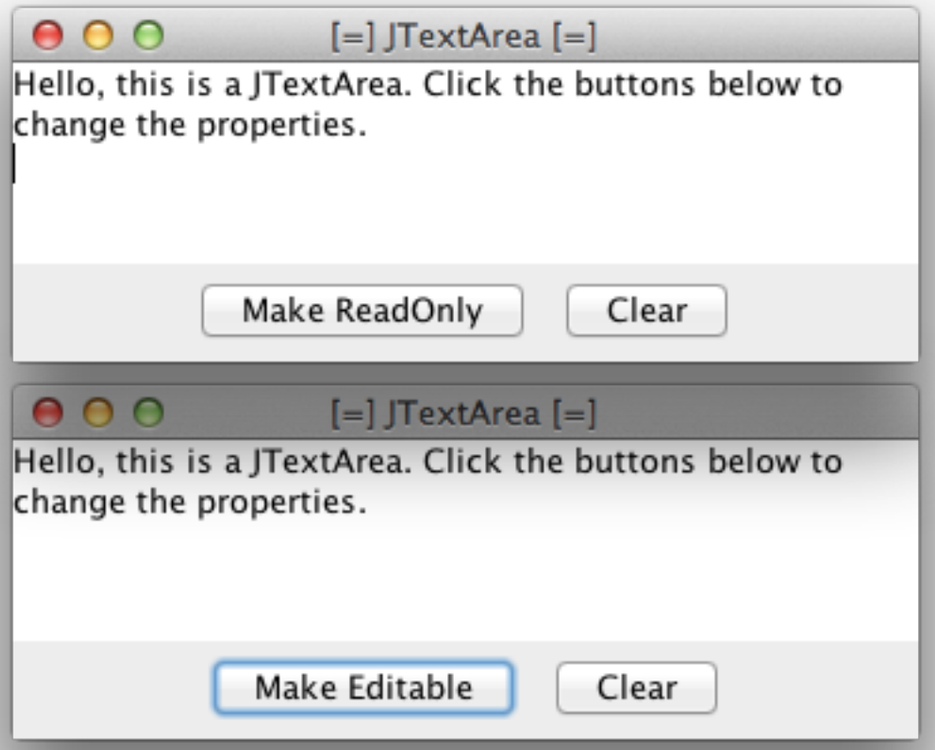
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
...
    private void initGUI() {
        ...
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

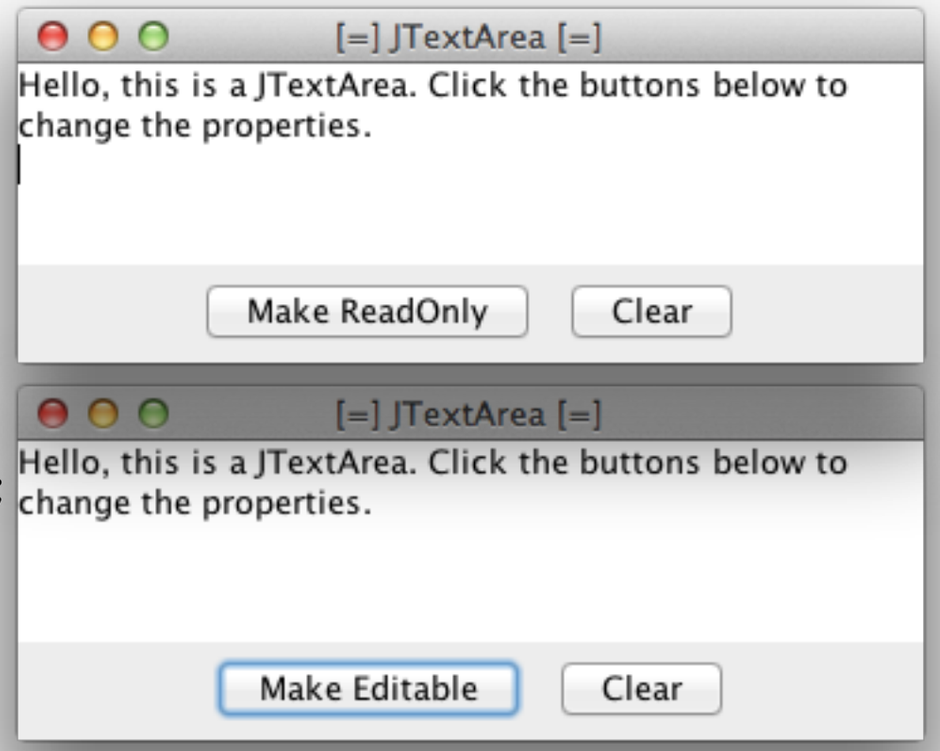
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
...
    private void initGUI() {
        ...
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

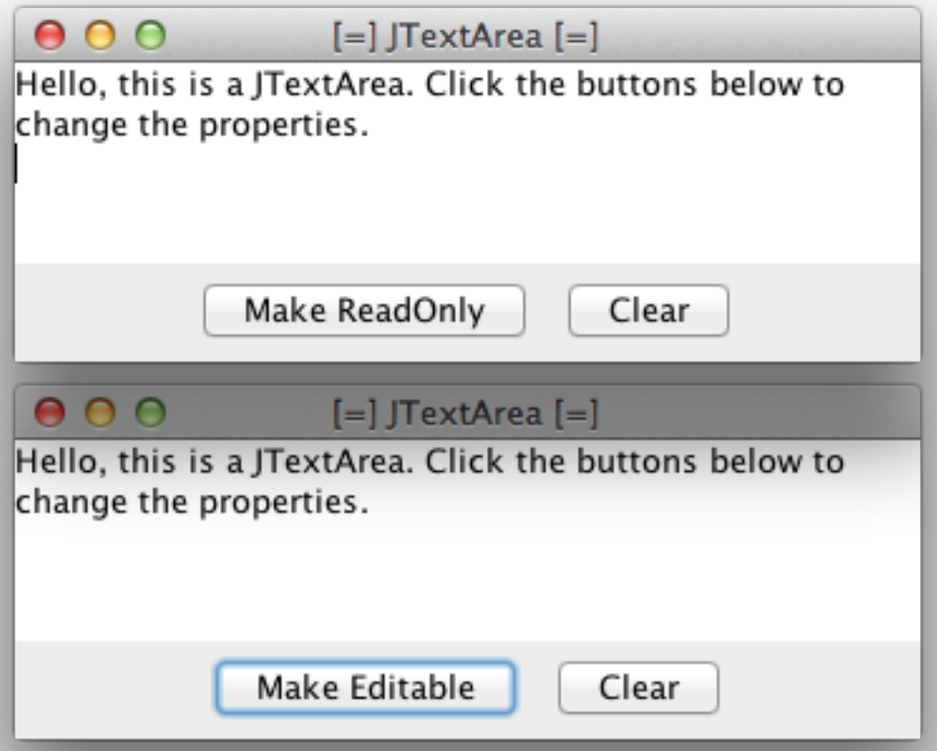
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  ...
    private void initGUI() {
      ...
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        ...
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

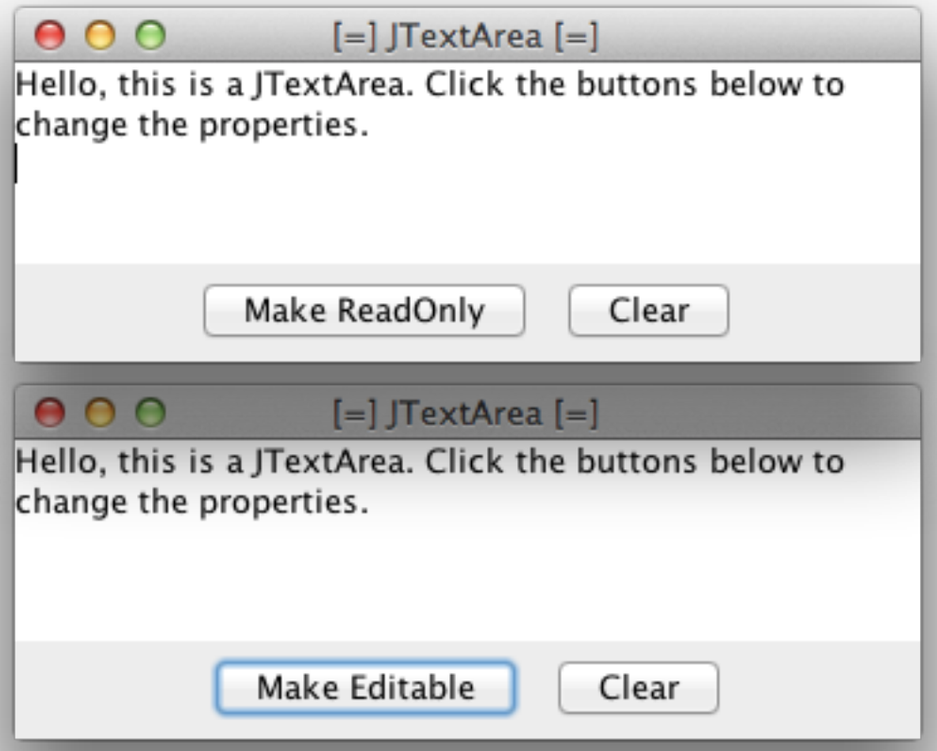
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  …
    private void initGUI() {
      …
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

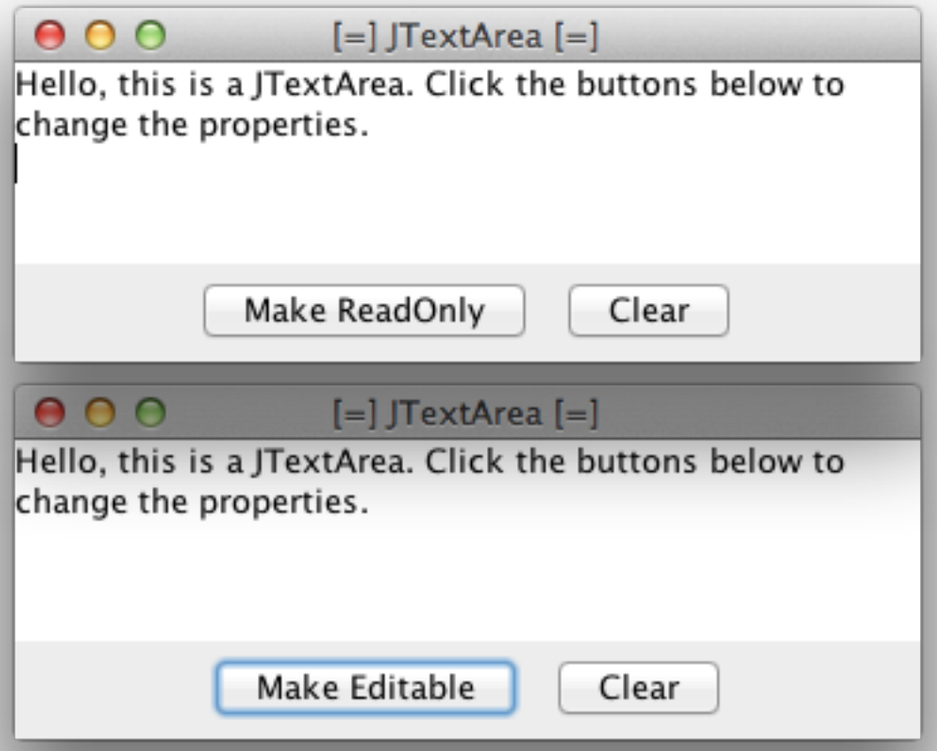
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  …
    private void initGUI() {
      …
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

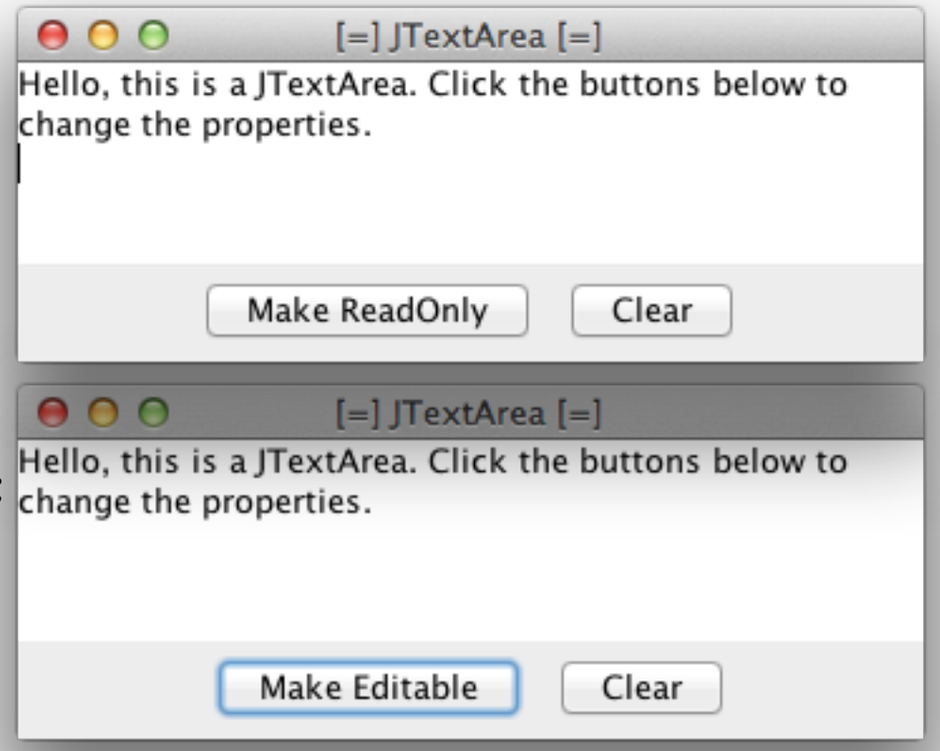
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  …
    private void initGUI() {
      …
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JTextArea

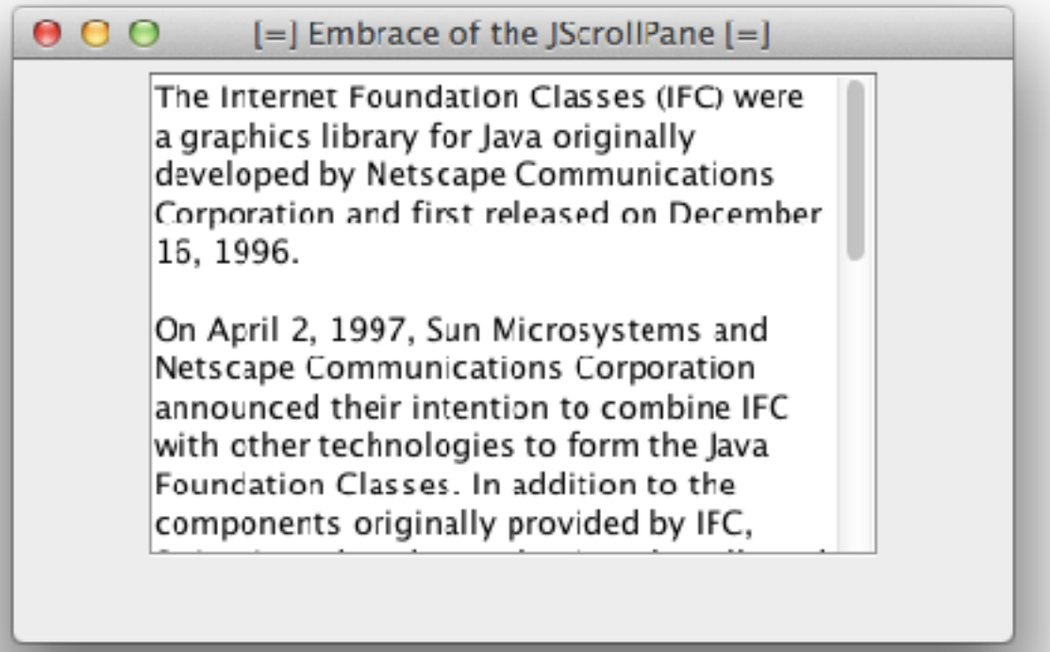
```
public class TextAreaExample extends JFrame implements ActionListener {

    JTextArea textArea;
    JButton readOnlyButton, clearButton;
  …
    private void initGUI() {
      …
        textArea = new JTextArea(5, 30);
        textArea.setEditable(true);
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);
        mainPanel.add(textArea, BorderLayout.CENTER);
        …
    }

    public void actionPerformed(ActionEvent e) {
        if ( e.getSource() == clearButton ) {
            textArea.setText("");
        } else if ( e.getSource() == readOnlyButton ) {
            boolean editable = textArea.isEditable();
            textArea.setEditable(!editable);
            if (editable)
                readOnlyButton.setText("Make Editable");
            else
                readOnlyButton.setText("Make Read Only");
        }
    }
```



see: examples.swing.misc.TextAreaExample

JScrollPane

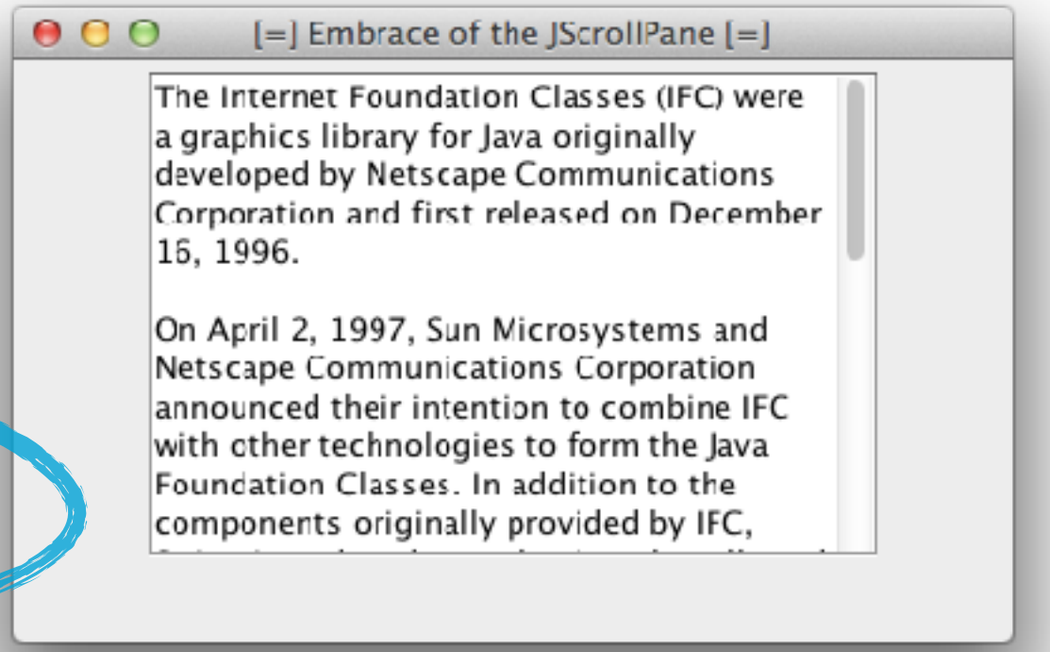
```
private void initGUI () {
    ...
      String story = "The ....";

      JTextArea storyArea = new JTextArea(story);
      storyArea.setEditable(false);
      storyArea.setLineWrap(true);
      storyArea.setWrapStyleWord(true);

      JScrollPane area =
            new JScrollPane(storyArea,
                        JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED,
                          JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);

      area.setPreferredSize(new Dimension(300, 200));
      mainPanel.add(area);

}
```



see: examples.swing.misc.ScrollPaneExample

JScrollPane

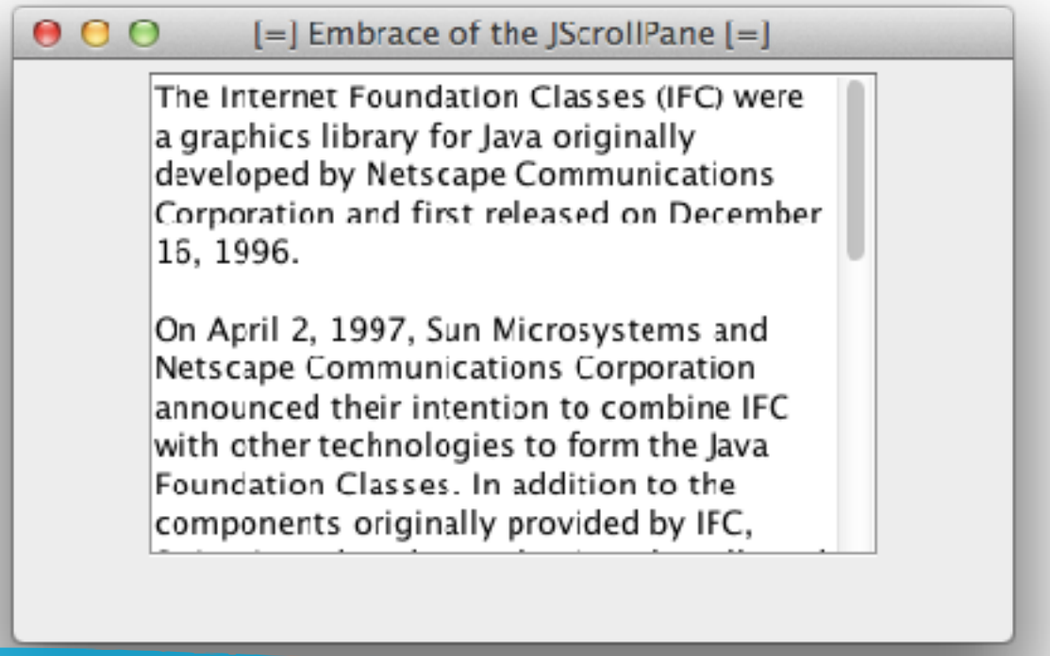
```
private void initGUI () {
    ...
      String story = "The ....";

      JTextArea storyArea = new JTextArea(story);
      storyArea.setEditable(false);
      storyArea.setLineWrap(true);
      storyArea.setWrapStyleWord(true);

      JScrollPane area =
            new JScrollPane(storyArea,
                        JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED,
                          JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);

      area.setPreferredSize(new Dimension(300, 200));
      mainPanel.add(area);

}
```



see: examples.swing.misc.ScrollPaneExample

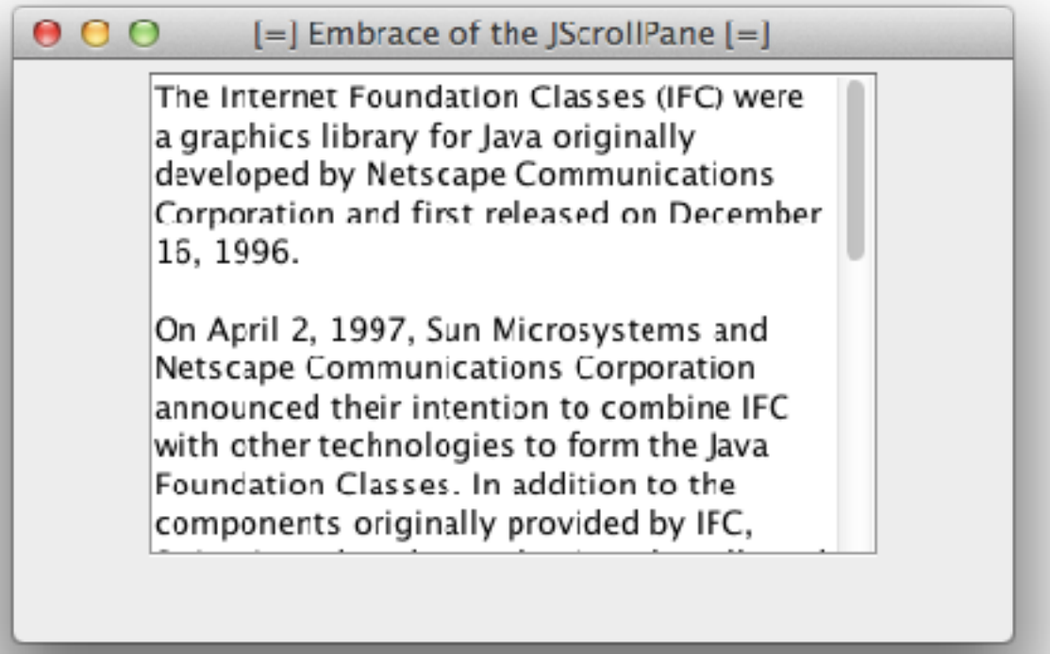
JScrollPane

```
private void initGUI () {
    ...
      String story = "The ....";

      JTextArea storyArea = new JTextArea(story);
      storyArea.setEditable(false);
      storyArea.setLineWrap(true);
      storyArea.setWrapStyleWord(true);

      JScrollPane area =
            new JScrollPane(storyArea,
                            JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED,
                              JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);

      area.setPreferredSize(new Dimension(300, 200));
      mainPanel.add(area);

}
```



see: examples.swing.misc.ScrollPaneExample

JScrollPane



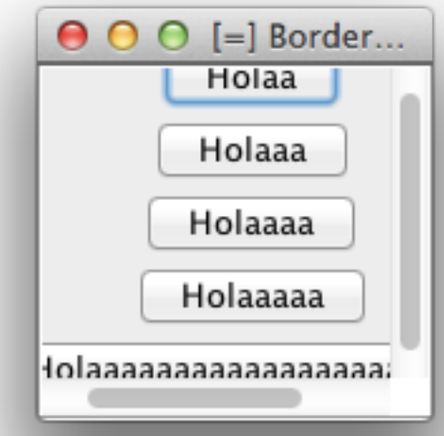

```
private void initGUI () {
    ...
      String story = "The ....";

      JTextArea storyArea = new JTextArea(story);
      storyArea.setEditable(false);
      storyArea.setLineWrap(true);
      storyArea.setWrapStyleWord(true);

      JScrollPane area =
            new JScrollPane(storyArea,
                        JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED,
                          JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);

      area.setPreferredSize(new Dimension(300, 200));
      mainPanel.add(area);

}
```

```
JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED
JScrollPane.VERTICAL_SCROLLBAR_ALWAYS
JScrollPane.VERTICAL_SCROLLBAR_NEVER

JScrollPane. HORIZONTAL_SCROLLBAR_AS_NEEDED
JScrollPane. HORIZONTAL_SCROLLBAR_ALWAYS
JScrollPane. HORIZONTAL_SCROLLBAR_NEVER
```

see: examples.swing.misc.ScrollPaneExample

JScrollPane with JPanel

```
private void initGUI() {

    JPanel mainPanel = new JPanel( new BorderLayout() );

    JPanel p = new JPanel();
    p.setLayout(new BoxLayout(p, BoxLayout.Y_AXIS));

    addButton("Holaa", p);
    addButton("Holaaa", p);
    addButton("Holaaaa", p);
    addButton("Holaaaaa", p);
    addButton("Holaaaaaaaaaaaaaaaaaaaaaaa", p);

    JScrollPane area =
    new JScrollPane (p,
                     JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED,
                         JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);
    mainPanel.add(area);
    ...

}
```



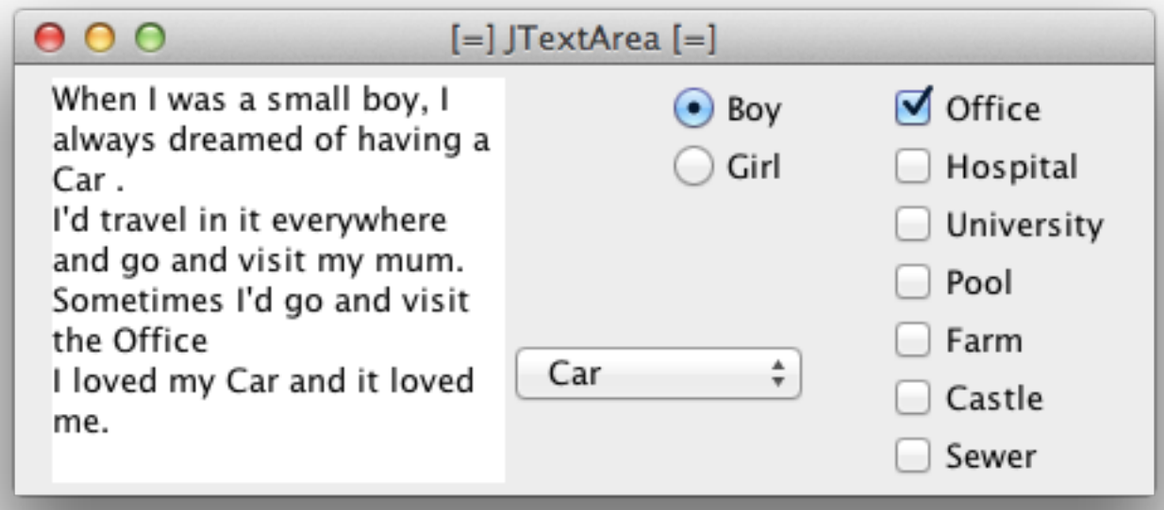
see: examples.swing.misc.ScrollPaneExample_1

JSeparator

```
private void initGUI() {

    JPanel mainPanel = new JPanel();
    mainPanel.setLayout(new BoxLayout(mainPanel, BoxLayout.X_AXIS));

    mainPanel.add(Box.createRigidArea(new Dimension(10, 0)));
    mainPanel.add(storyArea);
    mainPanel.add(new JSeparator(SwingConstants.VERTICAL));
    mainPanel.add(sexPanel);
    mainPanel.add(Box.createRigidArea(new Dimension(10, 0)));
    mainPanel.add(new JSeparator(SwingConstants.VERTICAL));
    mainPanel.add(Box.createRigidArea(new Dimension(10, 0)));
    mainPanel.add(cbPanel);
    mainPanel.add(Box.createRigidArea(new Dimension(10, 0)));
    ...
}
```

with separators

without separators



see: examples.swing.misc.SeparatorExample