AWT Control: Choice

- Used to create a *pop-up list items*.
- **Default constructor, Choice()** create empty list.
- For add item in list and select active item:
  - void add(String name)
  - void select(int *index)*
  - void select(String *name)*
- Each item in the list is a string that appears as a left-justified label in the order it is added to the **Choice object.**
- To determine selected item:
  - String getSelectedItem( )
  - int getSelectedIndex( )
  - String getItem(**int *index*)**

AWT Control: Choice (Example)

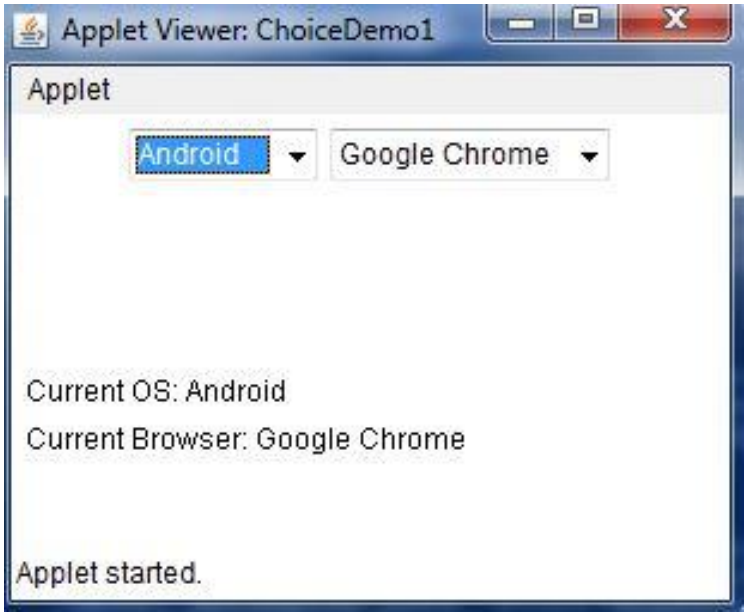
```java
//Demonstrate Choice Lists.
import java.awt.*;
import java.applet.*;
/*
<applet code="ChoiceDemo" width=300 height=180></applet>
*/
public class ChoiceDemo extends Applet
{
    Choice os, browser;
    public void init()
    {
        os=new Choice();
        browser=new Choice();

        //add items to os list
        os.add("Windows");
        os.add("Android");
        os.add("Solaris");
        os.add("Mac OS");

        //add items to browser list
        browser.add("Internet Explorer");
        browser.add("Mozilla Firefox");
        browser.add("Google Chrome");

        //add choice lists to window
        add(os);
        add(browser);
    }
}
```

AWT Control: Handling Choice Lists

- When **Choice** selected, an item event is generated.
- Implements the ***ItemListener*** **interface.**
- Interface defines the **itemStateChanged( ) method.**
- **ItemEvent** object is supplied as the argument to this method.

AWT Control: Handling Choice Lists (Example)

```
//Demonstrate Choice Lists.
import java.awt.*;import java.applet.*;
import java.awt.event.*;
/*<applet code="ChoiceDemo1" width=300 height=180></applet>*/
public class ChoiceDemo1 extends Applet implements ItemListener
{
    Choice os, browser;
    String msg="";
    public void init()
    {
        os=new Choice();
        browser=new Choice();
        //add items to os list
        os.add("Windows");
        os.add("Android");
        os.add("Solaris");
        os.add("Mac OS");
        //add items to browser list
        browser.add("Internet Explorer");
        browser.add("Mozilla Firefox");
        browser.add("Google Chrome");
        //add choice lists to window
        add(os);
        add(browser);
        //register to receive item events
        os.addItemListener(this);
        browser.addItemListener(this);
    }
    public void itemStateChanged(ItemEvent ie)
    {
        repaint();
    }
    //Display current selections
    public void paint(Graphics g)
    {
        msg="Current OS: ";
        msg+=os.getSelectedItem();
        g.drawString(msg, 6, 120);
        msg="Current Browser: ";
        msg+=browser.getSelectedItem();
        g.drawString(msg,6,140);
    }
}
```



AWT Control: List

- **List** class provides a *compact*, *multiple-choice*, scrolling *selection list*.

- **List** object can be constructed to *show any number of choices* in the visible window.

- In Choice **only one** item is shown.

- **Constructors**
  - **List()**
  - **List(int *numRows*)**
  - **List(int *numRows*, *boolean multipleSelect*)**

AWT Control: List

- **Following methods are used to add items:**
  - void add(**String *name*)**
  - void add(**String *name, int index*)**

- **For single selection items:**
  - String getSelectedItem( )
  - int getSelectedIndex( )

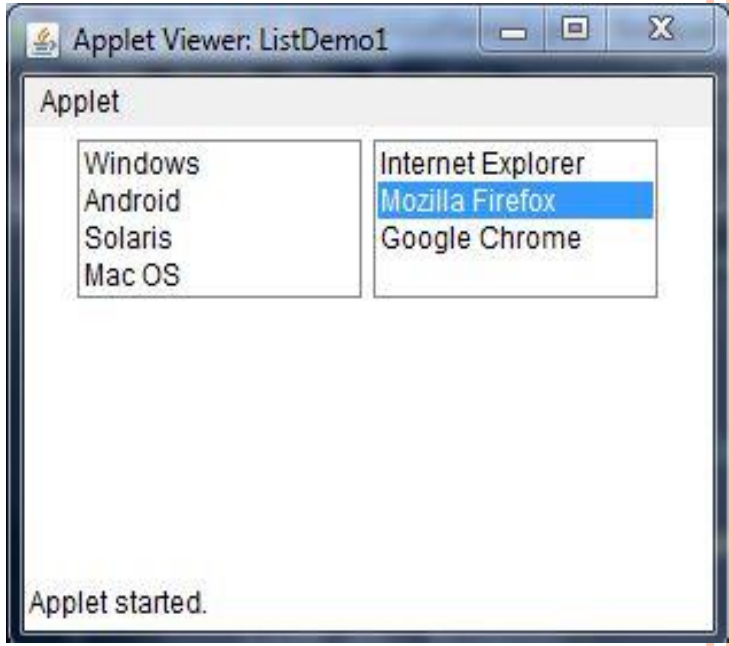
- **For Multi selection items:**
  - String[ ] getSelectedItems( )
  - int[ ] getSelectedIndexes( )

AWT Control: List

- **To retrieve item:**
  - String getItem(int *index)*
- **To get Item Count**
  - int getItemCount( )
- **Active Item**
  - void select(int *index)*

AWT Control: List (Example)

```
//Demonstrate Lists
import java.awt.*;
import java.applet.*;
/*
<applet code="ListDemo1" width=300 height=180></applet>
*/
public class ListDemo1 extends Applet
{
    List os, browser;
    public void init()
    {
        os=new List(4,true);
        browser=new List(4,false);

        //add items to os list
        os.add("Windows");
        os.add("Android");
        os.add("Solaris");
        os.add("Mac OS");

        //add items to browser list
        browser.add("Internet Explorer");
        browser.add("Mozilla Firefox");
        browser.add("Google Chrome");

        browser.select(1);

        add(os);
        add(browser);
    }
}
```



AWT Control: Handling Lists

- **Two types of event generated:**
  - For double clicked: **ActionEvent** generated.
  - For select and deselect item: **ItemEvent** generated.

- Implements **ActionListener** interface and **ItemListener**.

AWT Control: Handling Lists (Example)

```
//Demonstrate Lists with event handling
import java.awt.*; import java.awt.event.*;
import java.applet.*;
/*<applet code="ListDemo2" width=300 height=180></applet>*/
public class ListDemo2 extends Applet implements ActionListener
{
    List os, browser; String msg="";
    public void init()
    {
        os=new List(4,true);
        browser=new List(4,false);
        //add items to os list
        os.add("Windows");
        os.add("Android");
        os.add("Solaris");
        os.add("Mac OS");
        //add items to browser list
        browser.add("Internet Explorer");
        browser.add("Mozilla Firefox");
        browser.add("Google Chrome");

        browser.select(1);

        add(os);
        add(browser);
        //register to receive action events
        os.addActionListener(this);
        browser.addActionListener(this);
    }
    public void actionPerformed(ActionEvent ae)
    {
        repaint();
    }
    //Display Current Selections
    public void paint(Graphics g)
    {
        int idx[];

        msg="Current OS: ";
        idx=os.getSelectedIndexes();
        for(int i=0;i<idx.length;i++)
            msg += os.getItem(idx[i]) + " ";
        g.drawString(msg,6,120);
        msg="Current Browser: ";
        msg +=browser.getSelectedItem();
        g.drawString(msg, 6, 140);
    }
}
```