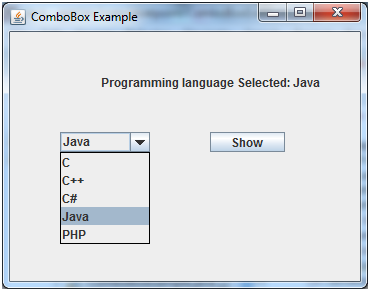
Parte 2:

utilización de JComboBox y JList

# JComboBox

Funcionamiento de un JComboBox

Un JComboBox es una lista desplegable de elementos que te permite seleccionar uno de ello, ejemplo:

La creación de un JComboBox se hace con *ayuda de la clase JComboBox perteneciente al* paquete javax.swing. esta clase tiene sobrecarga de constructores pero en esta definición nos enfocaremos solo en el constructor por defecto. La creación de un JComboBox se hace con la siguiente línea de código:

*JComboBox miComboBox = new JComboBox();*

Métodos más importantes de un JComboBox:

addItem( E ele ): permite agregar un elemento al combo box pide por parámetro un objeto del tipo que guarda el comboBox en nuestro caso será de tipo String.

*miComboBox.addItem( “Elemento 1” );*

removeItem( E ele ): permite eliminar un elemento del comboBox pide por parámetro un objeto del tipo que guarda el comboBox.

*miComboBox.removeItem ( “Elemento 1” );*

removeItemAt( int index ): permite eliminar un elemento del combo box pide por parámetro la posición del objeto .

*miComboBox.removeItem ( 0 );*

removeAllItems(): permite eliminar todos los elementos del comboBox

*miComboBox.removeAllItems ();*

getSelectedItem(): permite obtener el elemento seleccionado, este método devuelve un abjeto del tipo que guarda el comboBox en nuestro caso será un String, pero hay que hacerle un casting

*String itemSelected = (String)miComboBox.getSelectedItem ();*

getSelectedIndex(): permite obtener la posición del elemento seleccionado,

*int indexSelected = miComboBox.getSelectedIndex ();*

setSelectedItem ( E obj ): permite seleccionar un elemento.

*miComboBox. setSelectedItem (“Elemento 1”);*

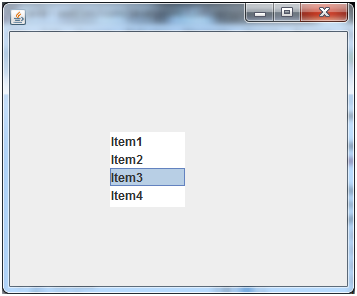
setSelectedIndex( int index ): permite seleccionar un elemento teniendo pasandole su posicion por parámetro.

*miComboBox.setSelectedIndex (0);*

getItemCount(): devuelve la cantidad de elementos contenida en el comboBox

*int indexSelected = miComboBox.getItemCount (0);*

# JList



Una JList es una lista de elementos en forma de menú, los cuales se pueden seleccionar (permite selección múltiple) e interactuar con ellos.

Para la crecion de un JList se necesita un ListModel que es el encargado de guardar los elementos de la JList

La creación de un JList y su respectivo ListModel se hace con las siguientes líneas de código (utilizaremos el modelo por defecto):

*DefaultListModel dlm = new DefaultListModel();*

*JList miLista = new JList ( dlm );*

Métodos Mas importantes de un JComboBox:

setModel( ListModel mod ): establece el modelo de lista

*miLista.setModel( dlm );*

addElement( E ele ): agrega un elemento al listModel

*dlm.addElement ( “Element 0” );*

remove( int index ): elimina el elemento del listModel en la posicion “index”

*dlm.remove ( 0 );*

removeAll( int index ): elimina todos los elementos del

*dlm.removeAll ( 0 );*

getSelectedIndex(): recupera la posición del elemento seleccionado de la lista

*int pos = miLista.getSelectedIndex ();*

getSelectedIndeces(): recupera las posiciones de los elementos seleccionados de la lista

*int[] pos = miLista.getSelectedIndices ();*