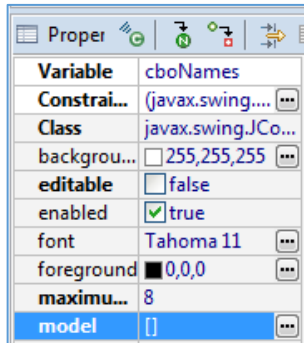
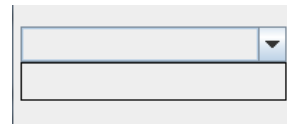


## Java WindowBuilder GUI Part 2

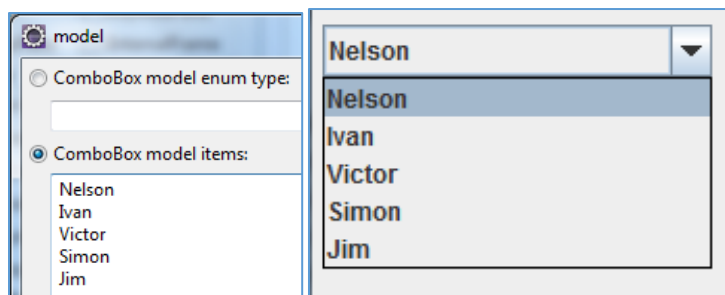
The follow notes show you how to populate a combobox and two listboxes (different methods)

A simple Combo box

Add a combobox component and change it to field



go to the model property and click on ...



This is what the WB has done for you

```
86      cboNames = new JComboBox();
87      cboNames.setModel(new DefaultComboBoxModel(new String[] {"Nelson", "Ivan", "Victor", "Simon", "Jim"}));
```

we are now going to code that ourselves with a tiny change.

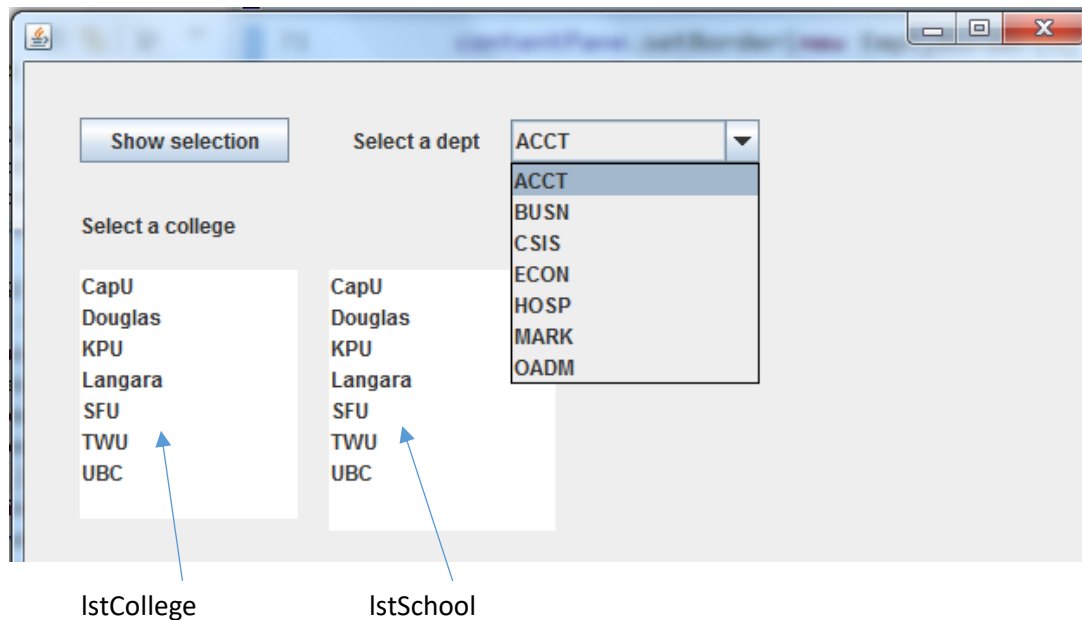
We have an array called deptNames and we want to use the array elements (sorted in alphabetical order) to populate the combobox cboDept.

```
79      cboDept = new JComboBox();
80      String [] deptNames = {"CSIS", "ACCT", "BUSN", "ECON", "HOSP", "OADM", "MARK"};
81      Arrays.sort(deptNames);
82
83      for (int x = 0; x <=6; x++)
84          cboDept.addItem(deptNames[x]);
```

Try this: `cboDept.setSelectedIndex(2);` to set the default selection. Why not on property?

To display the selected item on this combobox

```
52      cboDept.addActionListener(new ActionListener() {
53          public void actionPerformed(ActionEvent e) {
54              JOptionPane.showMessageDialog(null, cboDept.getSelectedItem());
55          }
56      });
```



We have another array called college. To populate a listbox **lStCollege** with this array's elements, we first need to instantiate a DLM object (line 88): collegeNames

We then use the addElement method to add items into this collegeNames object.

To complete the listbox with the collegeNames object.

```

86     String [] college = {"Douglas", "SFU", "UBC", "KPU", "CapU", "Langara", "TWU"};
87
88     DefaultListModel<String> collegeNames = new DefaultListModel<>(); //need to import DLM
89     Arrays.sort(college);
90     for (int y= 0; y <= 6; y++)
91         collegeNames.addElement(college[y]);
92
93     lStCollege = new JList(collegeNames);

```

To display what you have selected on both the combo box (SelectedItem) and listbox (SelectedValue)

```

58     btnShowSelection.addActionListener(new ActionListener() {
59         public void actionPerformed(ActionEvent e) {
60             String dept = cboDept.getSelectedItem().toString();
61             String school = lStCollege.getSelectedValue().toString();
62             lblMsg.setText("You have selected " + dept + " @ " + school);
63         }
64     });

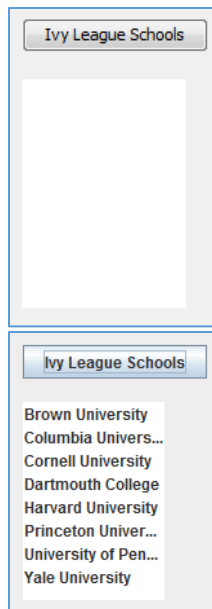
```

Another way to load the array elements to a listbox. Compare 93 105

```

101     DefaultListModel schoolNames = new DefaultListModel();
102     for (int y = 0; y <=6; y++)
103         schoolNames.addElement(college[y]);
104
105     lStSchool.setModel(schoolNames);

```



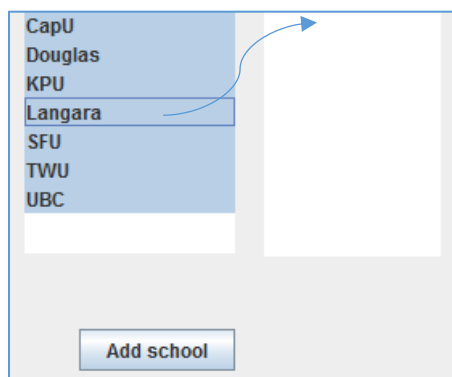
How about loading a data file with names onto a listbox?

```

70 btnIvy.addActionListener(new ActionListener() {
71     public void actionPerformed(ActionEvent e) {
72         DefaultListModel<String> elite = new DefaultListModel<>();
73         try
74         {
75             File myFile = new File("c:\\temp\\ivyLeague.txt");
76             Scanner myData = new Scanner(myFile);
77             while (myData.hasNext())
78             {
79                 elite.addElement(myData.nextLine().toString());
80             }
81             myData.close();
82             lstIvyLeague.setModel(elite);
83         }
84
85         catch (Exception ex)
86         {
87             System.out.println(ex.getMessage());
88         }
89     }
90 }

```

What about moving an item from one JList to another?



Note that Langara is moved from the left to the right

```

42 private String[] newSchools = new String[10];
43 private int count=0;

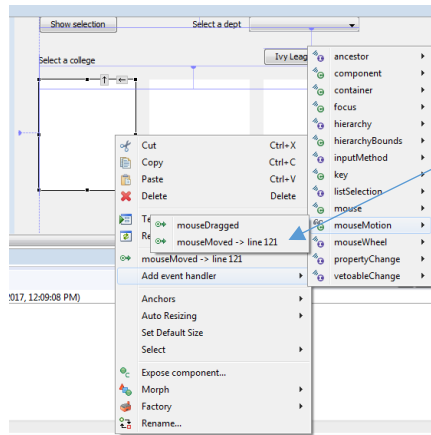
```

```

96 btnAdd.addActionListener(new ActionListener() {
97     public void actionPerformed(ActionEvent arg0) {
98         int toRemove=0;
99         String picked = lstSchool.getSelectedValue().toString();
100         toRemove = lstSchool.getSelectedIndex();
101
102         int ulimit = count;
103         newSchools[count]=picked; //store the picked items into array
104         count++;
105
106         DefaultListModel newSchool = new DefaultListModel<>();
107         for (int x = 0; x<=ulimit; x++)
108             newSchool.addElement(newSchools[x]); //put array into the new list
109
110         lstIvyLeague.setModel(newSchool);
111
112         DefaultListModel listRemoval = (DefaultListModel) lstSchool.getModel();
113
114         listRemoval.remove(toRemove); //remove the item from the original list
115     }
116 }

```

Something interesting. Right click on the right listbox, mouseMotion. You see somehow my screen capture shown line 121 because I have done the coding already. See next page, line 124.

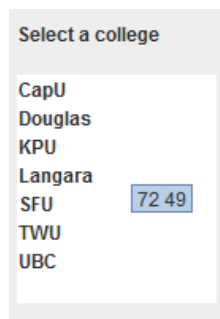


mouseMoved -> line 121  
mouseMoved -> line 121

```

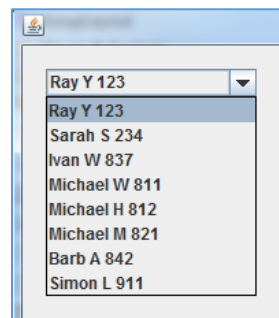
121     lstCollege.addMouseMotionListener(new MouseMotionAdapter() {
122         @Override
123         public void mouseMoved(MouseEvent arg0) {
124             → lstCollege.setToolTipText(arg0.getX() + " " + arg0.getY());
125         }
126     });

```



in this case x is 72, y is 49. 0,0 is top left corner.

How to extract the id?



```

69     cboStudents.addActionListener(new ActionListener() {
70         public void actionPerformed(ActionEvent arg0) {
71             String [] col = cboStudents.getSelectedItems().toString().split(" ");
72             String id = col[2];
73             JOptionPane.showMessageDialog(null, id);
74         }
75     });

```

