

Py

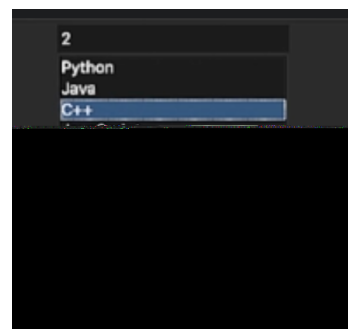
list box #2

- The list box methods are as follows

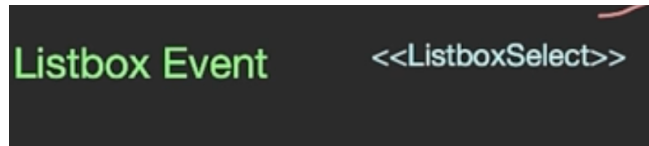
Listbox Methods

<code>activate(index)</code>	<code>see(index)</code>
<code>curselection()</code>	
<code>insert(index, *elements)</code>	<code>selection_clear(first, last=None)</code>
<code>get(first, last=None)</code>	<code>selection_includes(index)</code>
<code>delete(first, last=None)</code>	<code>selection_set(first, last=None)</code>
<code>itemconfig(index, option=value, ...)</code>	<code>size()</code>
<code>itemcget(index, option)</code>	

- `activate(index)` it is used to give the index value of the selected items, it can be a single or multiple items, when multiple items are selected it gives the value in the form of a tuple
- Look at the example output below



- Whenever you are clicking on an item the event that is generated is an event called <<ListboxSelect>> event



- This is a special extended event
- Its declared as shown below

```
def myhandler(e):  
    var.set(lst1.curselection())  
  
win = Tk()  
win.geometry('600x400')  
  
var = StringVar()  
ent1 = Entry(win, textvariable=var)  
ent1.pack()  
  
lst1 = Listbox(win, selectmode=EXTENDED)  
lst1.insert(0, 'Python')  
lst1.insert(1, 'Java')  
lst1.insert(2, 'C++')  
lst1.insert(3, 'JavaScript')  
lst1.insert(4, 'Ruby')  
lst1.insert(5, 'C Lang')  
lst1.insert(6, 'PHP')  
lst1.insert(7, 'C#')  
  
lst1.pack()  
  
btn1 = Button(win, text='Click Me')  
btn1.pack()  
  
lst1.bind('<<ListboxSelect>>', myhandler)  
  
win.mainloop()
```

- Hence by this we can handle event upon the widgets