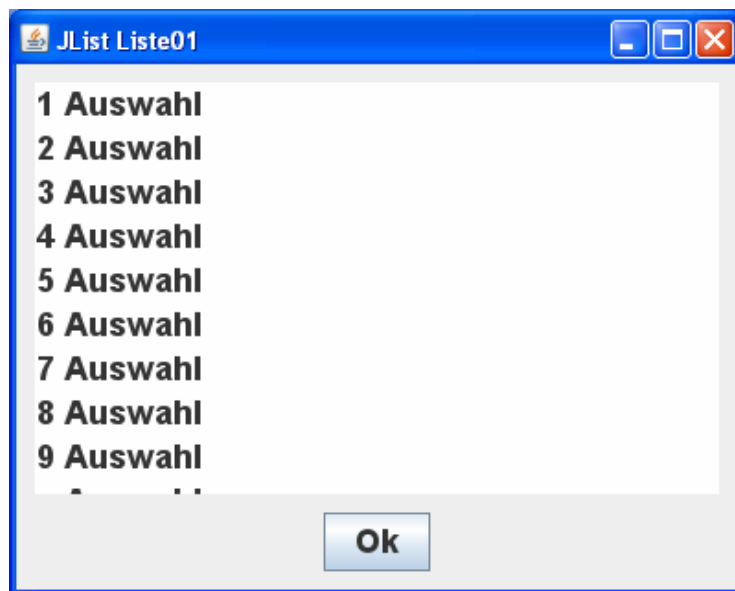


Beschreibung der Java-Dateien

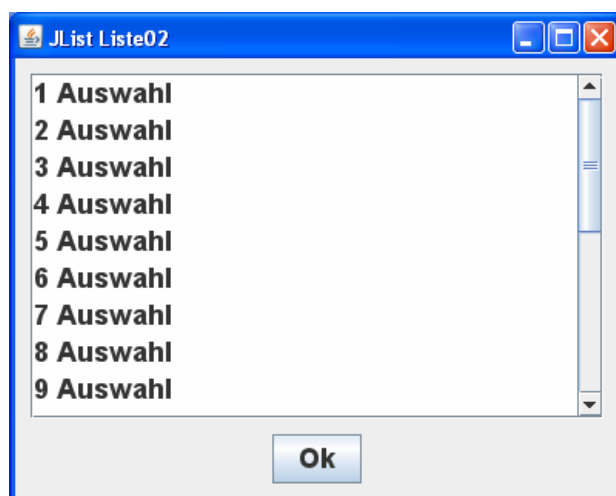
Liste01.java

Liste mit Array
ohne JScrollPane
Scrollen nicht möglich
GUI-Element mit GridBagLayout



Liste02.java

Liste mit Array
mit JScrollPane
Scrollen nun möglich
GUI-Element mit GridBagLayout
Ohne Action_proc



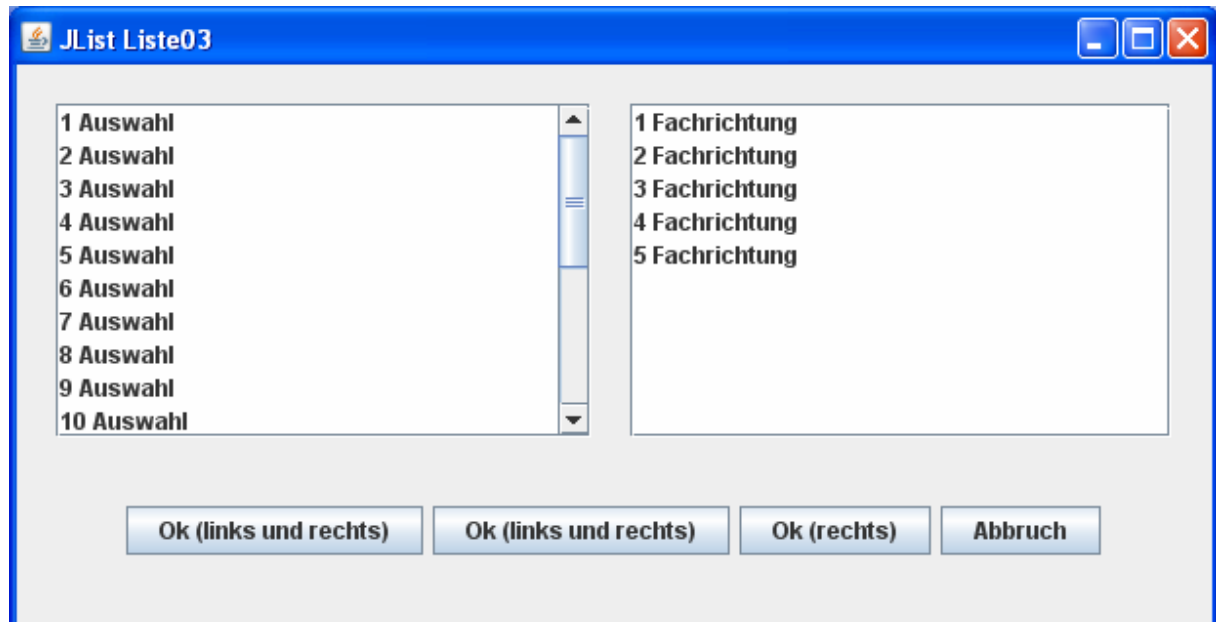
Liste03.java

Mit ActionListener

Die ersten beiden Schalter liegen auf einem Event

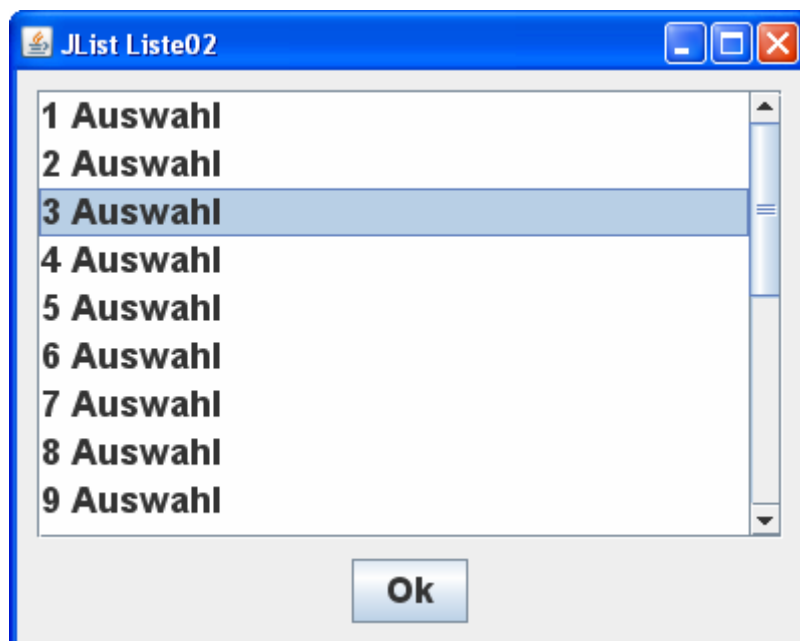
Rechte Liste mit multi-select

// Abfrage , Mehrfach Selection



Liste04.java

Liste mit Vector



```
import java.util.Vector;
```

```

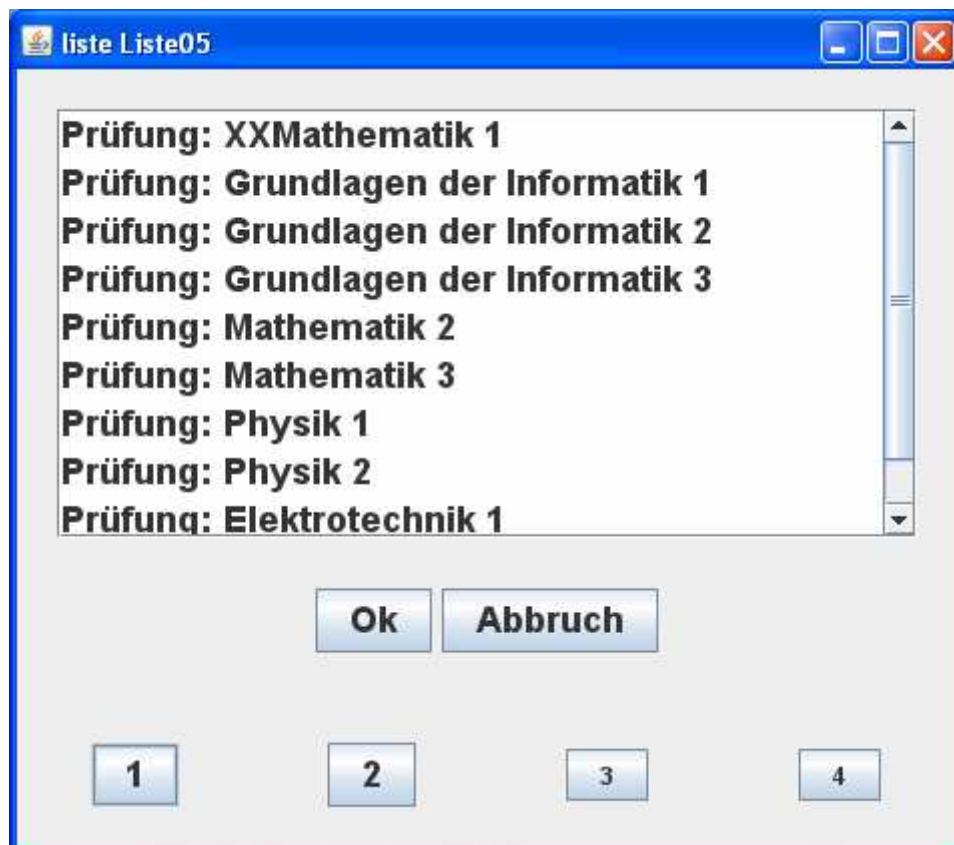
JList jlist;
JScrollPane scroll;
Vector items=new Vector();

// Vector für die JList
for (int i=0; i<20; i++) {
    items.add( Integer.toHexString(i+1)+" Auswahl" );
}
jlist = new JList(items);
scroll = new JScrollPane(jlist);
this.getContentPane().add(scroll, ....

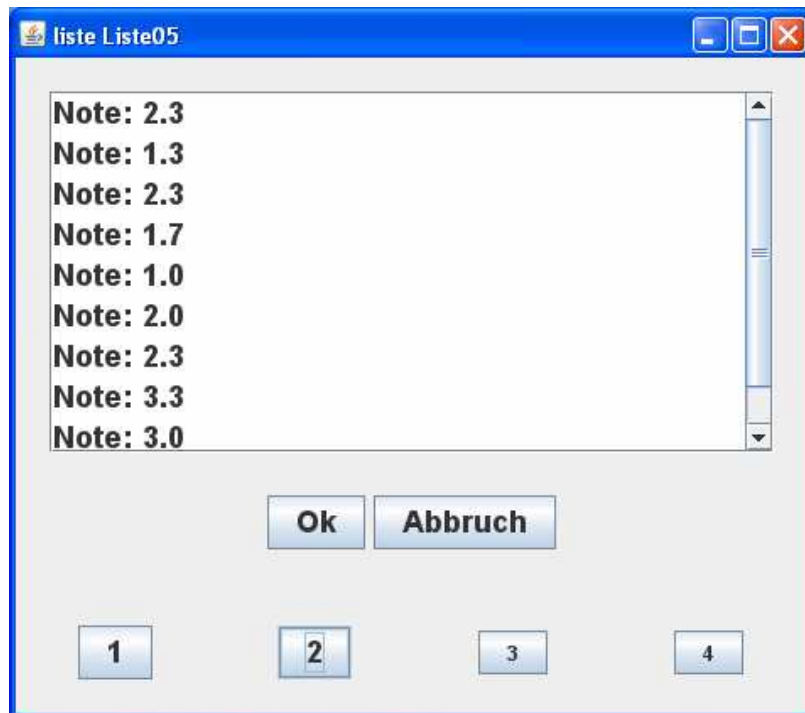
```

Liste05.java

Liste mit Vector
 Methode setData
 Eigene Klasse CPruefung mit toString !!!!!!!!!!!
 Action_proc Close
 mit JScrollPane
 GUI-Elemente mit GridBagLayout
 Action Proc (Abbruch)
 Ausgabe gesteuert über Schalter



Schalter 1



Schalter 2

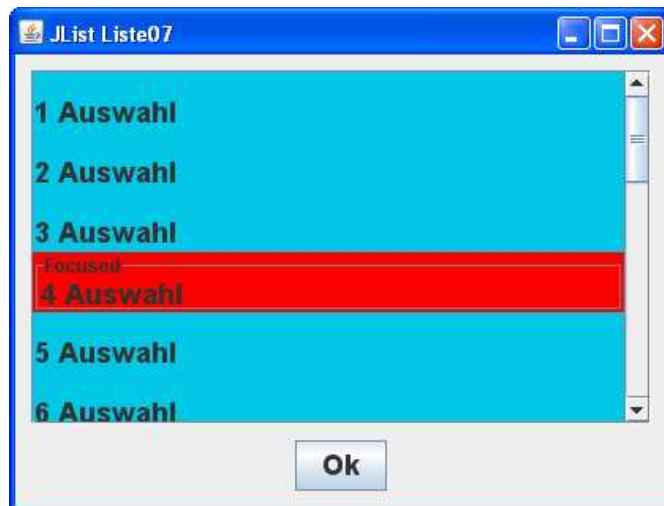
Liste06.java

Liste mit Vector (Strings)
 mit JScrollPane
 GUI-Elemente mit GridBagLayout
 Action Proc (Abbruch)
 Action_proc (Insert, Delete, Edit, Close)



Liste07.java

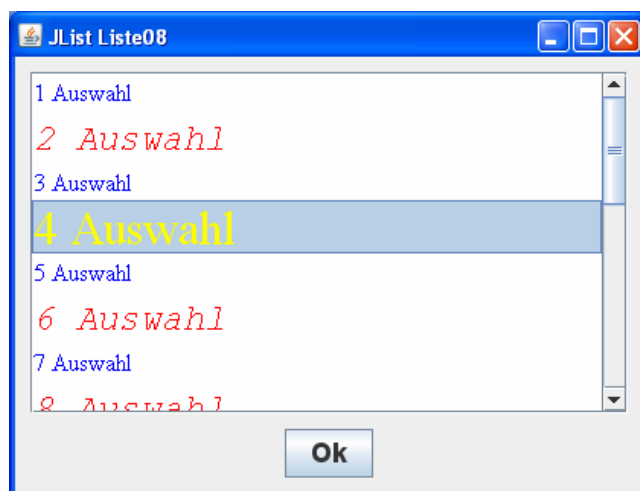
Liste mit Vector (Strings)
mit JScrollPane
GUI-Elemente mit GridBagLayout
Action Proc (Abbruch)
Liste mit cellRenderer und setFocus



Liste08.java

Liste mit Vector (Strings)
Eigene Klasse !!!!!!!!!!!!!!!
mit JScrollPane
GUI-Elemente mit GridBagLayout
Action Proc (Abbruch)
Liste mit cellRenderer und setFocus

Unterschiedliche Schriften pro Element



Liste09.java

zwei Listen mit Vector (Strings)

Beim Ok wird mittels getType() unterschieden, statt instanceof
Benutzt einen Interface

