

GUIMain.java

```
1 /*****
2  * @author Øystein
3  * GUIMain.class
4  *
5  * Hovedvinduet for å utføre endringer på studenter.
6  *****/
7 import java.awt.BorderLayout;
8 import java.awt.event.ActionEvent;
9 import java.awt.event.MouseEvent;
10 import java.awt.event.MouseListener;
11 import java.awt.event.WindowEvent;
12 import java.awt.event.WindowListener;
13 import javax.swing.DefaultListModel;
14 import javax.swing.JButton;
15 import javax.swing.JList;
16 import javax.swing.JScrollPane;
17
18 public class GUIMain extends Dialog implements MouseListener,
19     WindowListener{
20
21     //Main
22     //Vis valg (legg til studenter ELLER lag lister)
23     //Hvis legg til studenter
24     //--Spør om informasjon, lag nytt studentobjekt
25     //Hvis lag lister
26     //--Spør hva slags liste, så skriv ut.
27
28     private Gruppe gruppe = null;
29
30     private JList listStudenter = null;
31     private DefaultListModel listModel = new DefaultListModel();
32     private JScrollPane scrollStudenter = null;
33
34     private JButton btnRedigerElev = new JButton("Rediger elev og
35     oppgaver");
36     private Main m = null;
37
38     public GUIMain(Gruppe grp, Main m){
39         gruppe = grp;
40         this.m = m;
41         setup();
42
43         this.setVisible(true);
44     }
45 }
```

GUIMain.java

```
44
45     protected void setup(){ //Ordner vinduet klart til
    førstegangsvisning
46         //Kjører setup-metoden fra Dialog-klassen (som denne
    klassen arver fra)
47         super.setup();
48         this.setTitle("Studentadministrasjon");
49         this.setSize(320, 240);
50
51         /*Legger til en vinduslytter (Dette for å vite når vinduet
52         lukkes, slik at visMeny() fra Main kan kalles opp igjen.*/
53         this.addWindowListener(this);
54
55         //Gjør klar listen med studenter
56         updateListe();
57         listStudenter = new JList(listModel);
58         //Aktiverer en scrollbar dersom det trengs til lista.
59         scrollStudenter = new JScrollPane(listStudenter);
60
61         //Legg komponenter og paneler til i dialogvinduet
62         this.add(scrollStudenter, BorderLayout.CENTER);
63         this.add(btnRedigerElev, BorderLayout.SOUTH);
64
65         //Legg til lyttere etter knappetrykk og museklikk
66         btnRedigerElev.addActionListener(this);
67         listStudenter.addMouseListener(this);
68
69         //Sentrer dialogen på PC-skjermen
70         centerScreen();
71     }
72
73     //Returnerer studentobjektet som tilhører studenten som er valgt
    i lista.
74     private Student getValgteStudent(){
75         return gruppe.hentStudenterAsVector().elementAt(
76             listStudenter.getSelectedIndex());
77     }
78
79     private void updateListe(){//Oppdaterer innholdet i listboksen
80         Student[] stud = gruppe.hentStudenterAsArray();
81
82         listModel.clear(); //tøm lista
83
84         //Legg til alle studenter i lista
85         for (int i = 0; i < stud.length; i++) {
```

GUIMain.java

```
86         listModel.addElement(stud[i]);
87     }
88
89 }
90
91 //Tar hånd om hva som skjer når noe blir klikket på
92 public void actionPerformed(ActionEvent ae) {
93
94     if (ae.getSource() == btnRedigerElev){
95         //Hvis knappen "btnRedigerElev" klikkes
96         //Vis et vindu for redigering av studenten som er valgt
97         new GUIStudent(gruppe, getValgteStudent());
98         Filbehandling.lagreGruppe(gruppe); //Lagre
99         updateListe(); //oppdater lista (i tilfelle
        navnendringer etc.).
100     }
101 }
102
103
104
105 public void mouseClicked(MouseEvent me) { //Hvis brukeren
        dobbeltklikker
106     if (me.getClickCount() == 2) {
107         //Vis et vindu for redigering av studenten som har
        blitt klikket.
108         new GUIStudent(gruppe, getValgteStudent());
109         Filbehandling.lagreGruppe(gruppe); //Lagre
110         updateListe(); //oppdater lista (i tilfelle
        navnendringer etc.).
111     }
112 }
113
114 public void windowClosed(WindowEvent arg0) {
115     // TODO Auto-generated method stub
116     m.visMeny();
117 }
118
119 /*Under her er alle (ubrukte) metodene som kreves
120 av ActionListener og WindowsListener*/
121
122 public void mouseEntered(MouseEvent arg0) {
123 }
124
125 public void mouseExited(MouseEvent arg0) {
126 }
```

GUIMain.java

```
127
128 public void mousePressed(MouseEvent arg0) {
129 }
130
131 public void mouseReleased(MouseEvent arg0) {
132 }
133
134 @Override
135 public void windowActivated(WindowEvent arg0) {
136     // TODO Auto-generated method stub
137 }
138
139 @Override
140 public void windowClosing(WindowEvent arg0) {
141     // TODO Auto-generated method stub
142 }
143
144 }
145
146 @Override
147 public void windowDeactivated(WindowEvent arg0) {
148     // TODO Auto-generated method stub
149 }
150
151 }
152
153 @Override
154 public void windowDeiconified(WindowEvent arg0) {
155     // TODO Auto-generated method stub
156 }
157
158 @Override
159 public void windowIconified(WindowEvent arg0) {
160     // TODO Auto-generated method stub
161 }
162
163 }
164
165 @Override
166 public void windowOpened(WindowEvent arg0) {
167     // TODO Auto-generated method stub
168 }
169
170 }
171
```