

# GraphVisualizer.java

```

1 package lehmanrulesgraphsdisplay;
2
3 import java.util.ArrayList;
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18 public class GraphVisualizer {
19     private History h;
20
21     public GraphVisualizer(History h){
22         this.h = h;
23     }
24
25     public ChartPanel []firstRuleGraphGenerator(){
26         ChartPanel []panels= new ChartPanel[2];
27         DefaultCategoryDataset objDataset = new DefaultCategoryDataset();
28         ArrayList<VersionInfo> versionlist = h.getVersions();
29         int i;
30         for(i=0; i<versionlist.size(); i++){
31             double dops = versionlist.get(i).getopsChanges().getNumOfChanges();
32             String xaxis = versionlist.get(i).getID();
33             objDataset.setValue(dops,"Operations", xaxis);
34         }
35         for(i=0; i<versionlist.size(); i++){
36             double dstructs = versionlist.get(i).getstructsChanges().getNumOfChanges();
37             String xaxis = versionlist.get(i).getID();
38             objDataset.setValue(dstructs,"Structs", xaxis);
39         }
40         JFreeChart objChart = ChartFactory.createBarChart(
41             "Changes Bar Chart",
42             "Version ID",
43             "Changes",
44             objDataset,
45             PlotOrientation.VERTICAL,
46             true,
47             true,
48             false
49         );
50         panels[0] = new ChartPanel(objChart);
51         DefaultCategoryDataset objDataset2 = new DefaultCategoryDataset();
52         int [][]table;
53         if(versionlist.size()<=0){
54             return null;
55         }
56         int first= versionlist.get(0).yearOfDate();
57         int last = versionlist.get(versionlist.size()-1).yearOfDate();
58         int size = last-first+1;
59         table = new int[size][2];
60         for(i=0; i<size; i++){
61             table[i][0] = first + i;
62             table[i][1] = 0;
63         }
64
65         for(i=0; i<versionlist.size(); i++){
66             int year = versionlist.get(i).yearOfDate();
67             for(int j=0; j<size; j++){
68                 if(table[j][0]==year){
69                     table[j][1]++;
70                     break;
71                 }
72             }
73         }
74

```

# GraphVisualizer.java

```

75     for(i=0; i<size; i++){
76         String xaxis = ""+(table[i][0]-2000);
77         objDataset2.setValue(table[i][1],"Year Versions", xaxis);
78     }
79 }
80 JFreeChart objChart2 = ChartFactory.createBarChart(
81     "Version per Year Bar Chart",
82     "Version ID",
83     "Versions",
84     objDataset2,
85     PlotOrientation.VERTICAL,
86     true,
87     true,
88     false
89 );
90 panels[1] = new ChartPanel(objChart2);
91
92 return panels;
93
94 }
95
96 public ChartPanel[] secondRuleGraphGenerator()
97 {
98     ChartPanel []panels= new ChartPanel[2];
99     DefaultCategoryDataset objDataset = new DefaultCategoryDataset();
100     ArrayList<VersionInfo> versionlist = h.getVersions();
101     int i;
102     for(i=0; i<versionlist.size(); i++){
103         double dops = versionlist.get(i).getopsMetrics().getComplexity();
104         String xaxis = versionlist.get(i).getID();
105         objDataset.setValue(dops,"Operations", xaxis);
106     }
107     for(i=0; i<versionlist.size(); i++){
108         double dstructs = versionlist.get(i).getstructsMetrics().getComplexity();
109         String xaxis = versionlist.get(i).getID();
110         objDataset.setValue(dstructs,"Structs", xaxis);
111     }
112     JFreeChart objChart = ChartFactory.createLineChart(
113         "Complexity Line Chart",
114         "Version ID",
115         "Complexity",
116         objDataset,
117         PlotOrientation.VERTICAL,
118         true,
119         true,
120         false
121     );
122     panels[0] = new ChartPanel(objChart);
123     DefaultCategoryDataset objDataset2 = new DefaultCategoryDataset();
124     for(i=0; i<versionlist.size(); i++){
125         double dops = versionlist.get(i).getopsChanges().getMaintenanceActivities();
126         String xaxis = versionlist.get(i).getID();
127         objDataset2.setValue(dops,"Operations", xaxis);
128     }
129     for(i=0; i<versionlist.size(); i++){
130         double dstructs =
131         versionlist.get(i).getstructsChanges().getMaintenanceActivities();
132         String xaxis = versionlist.get(i).getID();
133         objDataset2.setValue(dstructs,"Structs", xaxis);
134     }
135     JFreeChart objChart2 = ChartFactory.createBarChart(

```

# GraphVisualizer.java

```

135         "Maintenance Bar Chart",
136         "Version ID",
137         "Maintenance",
138         objDataset2,
139         PlotOrientation.VERTICAL,
140         true,
141         true,
142         false
143     );
144     panels[1] = new ChartPanel(objChart2);
145     return panels;
146 }
147
148
149
150 public ChartPanel[] thirdRuleGraphGenerator()
151 {
152     ChartPanel []panels= new ChartPanel[2];
153     DefaultCategoryDataset objDataset = new DefaultCategoryDataset();
154     ArrayList<VersionInfo> versionlist = h.getVersions();
155     int i;
156     for(i=0; i<versionlist.size(); i++){
157         double dops = versionlist.get(i).getopsMetrics().getGrowthRate();
158         String xaxis = versionlist.get(i).getID();
159         objDataset.setValue(dops,"Operations", xaxis);
160     }
161     JFreeChart objChart = ChartFactory.createLineChart(
162         "Growth Rate Line Chart",
163         "Version ID",
164         "Growth Rate",
165         objDataset,
166         PlotOrientation.VERTICAL,
167         true,
168         true,
169         false
170     );
171     panels[0] = new ChartPanel(objChart);
172     DefaultCategoryDataset objDataset2 = new DefaultCategoryDataset();
173     for(i=0; i<versionlist.size(); i++){
174         double dstructs = versionlist.get(i).getstructsMetrics().getGrowthRate();
175         String xaxis = versionlist.get(i).getID();
176         objDataset2.setValue(dstructs,"Structs", xaxis);
177     }
178     JFreeChart objChart2 = ChartFactory.createLineChart(
179         "Growth Rate Bar Chart",
180         "Version ID",
181         "Growth Rate",
182         objDataset2,
183         PlotOrientation.VERTICAL,
184         true,
185         true,
186         false
187     );
188     panels[1] = new ChartPanel(objChart2);
189
190     return panels;
191 }
192
193 public ChartPanel[] fourthRuleGraphGenerator()
194 {
195     ChartPanel []panels= new ChartPanel[2];

```

# GraphVisualizer.java

```

196 DefaultCategoryDataset objDataset = new DefaultCategoryDataset();
197 ArrayList<VersionInfo> versionlist = h.getVersions();
198 int i;
199 for(i=0; i<versionlist.size(); i++){
200     double dops = versionlist.get(i).getopsMetrics().getWorkRate();
201     String xaxis = versionlist.get(i).getID();
202     objDataset.setValue(dops,"Operations", xaxis);
203 }
204 JFreeChart objChart = ChartFactory.createLineChart(
205     "Work Rate Line Chart",
206     "Version ID",
207     "Work Rate",
208     objDataset,
209     PlotOrientation.VERTICAL,
210     true,
211     true,
212     false
213 );
214 panels[0] = new ChartPanel(objChart);
215 DefaultCategoryDataset objDataset2 = new DefaultCategoryDataset();
216 for(i=0; i<versionlist.size(); i++){
217     double dstructs = versionlist.get(i).getstructsMetrics().getWorkRate();
218     String xaxis = versionlist.get(i).getID();
219     objDataset2.setValue(dstructs,"Structs", xaxis);
220 }
221 JFreeChart objChart2 = ChartFactory.createLineChart(
222     "Work Rate Bar Chart",
223     "Version ID",
224     "Work Rate",
225     objDataset2,
226     PlotOrientation.VERTICAL,
227     true,
228     true,
229     false
230 );
231 panels[1] = new ChartPanel(objChart2);
232 return panels;
233 }
234
235 public ChartPanel[] fifthRuleGraphGenerator()
236 {
237     return thirdRuleGraphGenerator();
238 }
239
240 public ChartPanel[] sixthRuleGraphGenerator()
241 {
242
243     ChartPanel []panels= new ChartPanel[2];
244     DefaultCategoryDataset objDataset = new DefaultCategoryDataset();
245     ArrayList<VersionInfo> versionlist = h.getVersions();
246     int i;
247     for(i=0; i<versionlist.size(); i++){
248         double dops = versionlist.get(i).getopsMetrics().getNumber();
249         String xaxis = versionlist.get(i).getID();
250         objDataset.setValue(dops,"Operations", xaxis);
251     }
252     JFreeChart objChart = ChartFactory.createLineChart(
253         "Work Rate Line Chart",
254         "Version ID",
255         "Work Rate",
256         objDataset,

```

# GraphVisualizer.java

```

257         PlotOrientation.VERTICAL,
258         true,
259         true,
260         false
261     );
262     panels[0] = new ChartPanel(objChart);
263     DefaultCategoryDataset objDataset2 = new DefaultCategoryDataset();
264     for(i=0; i<versionlist.size(); i++){
265         double dstructs = versionlist.get(i).getstructsMetrics().getNumber();
266         String xaxis = versionlist.get(i).getID();
267         objDataset2.setValue(dstructs, "Structs", xaxis);
268     }
269     JFreeChart objChart2 = ChartFactory.createLineChart(
270         "Work Rate Bar Chart",
271         "Version ID",
272         "Work Rate",
273         objDataset2,
274         PlotOrientation.VERTICAL,
275         true,
276         true,
277         false
278     );
279     panels[1] = new ChartPanel(objChart2);
280
281     return panels;
282 }
283
284 public ChartPanel[] seventhRuleGraphGenerator()
285 {
286     return null;
287 }
288
289 public ChartPanel[] eighthRuleGraphGenerator()
290 {
291     return null;
292 }
293 }
294

```