



Recherche dans DEJ avec Google

Rechercher



98. Des bibliothèques open source

Chapitre 98

Niveau :

Supérieur

L'écosystème Java est très riche en bibliothèques open source qui couvrent de très nombreux sujets. Leur utilisation permet d'être productif avec des solutions fiables et stables.

Ce chapitre contient plusieurs sections :

- [JFreeChart](#)
- [Beanshell](#)
- [Apache Commons](#)
- [Quartz](#)
- [JGoodies](#)
- [Apache Lucene](#)

98.1. JFreeChart

JFreeChart est une bibliothèque open source qui permet d'afficher des données statistiques sous la forme de graphiques. Elle possède plusieurs formats dont le camembert, les barres ou les lignes et propose de nombreuses options de configuration pour personnaliser le rendu des graphiques. Elle peut s'utiliser dans des applications standalone ou des applications web et permet également d'exporter le graphique sous la forme d'une image.

<http://www.jfree.org/jfreechart/>

La version utilisée dans cette section est la 0.9.18.

Pour l'utiliser, il faut télécharger le fichier jfreechart-0.9.18.zip et le décompresser. Son utilisation nécessite l'ajout dans le classpath des fichiers jfreechart-0.9.18.zip et des fichiers .jar présents dans le répertoire lib décompressé.

Les données utilisées dans le graphique sont encapsulées dans un objet de type Dataset. Il existe plusieurs sous-types de cette classe en fonction du type de graphique souhaité.

Un objet de type JFreechart encapsule le graphique. Une instance d'un tel objet est obtenue en utilisant une des méthodes de la classe ChartFactory.

Exemple : Un exemple avec un graphique en forme de camembert

```

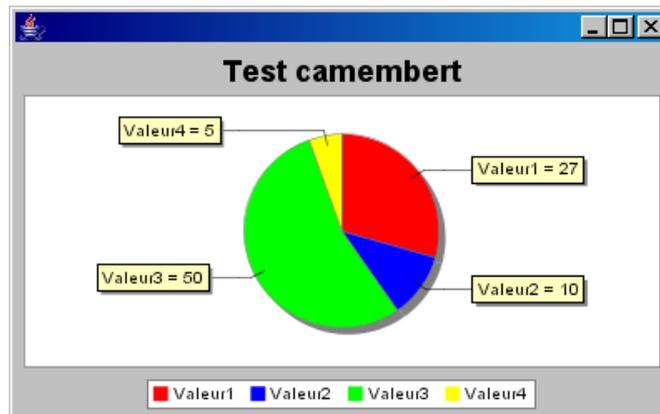
01. import java.awt.*;
02. import java.awt.event.*;
03. import javax.swing.*;
04. import org.jfree.chart.*;
05. import org.jfree.chart.plot.*;
06. import org.jfree.data.*;
07.
08. public class TestPieChart extends JFrame {
09.     private JPanel pnl;
10.
11.     public TestPieChart() {
12.         addWindowListener(new WindowAdapter() {
13.             public void windowClosing(WindowEvent e) {
14.                 dispose();

```

```

15.         System.exit(0);
16.     }
17. });
18. pnl = new JPanel(new BorderLayout());
19. setContentPane(pnl);
20. setSize(400, 250);
21.
22. DefaultPieDataset pieDataset = new DefaultPieDataset();
23. pieDataset.setValue("Valeur1", new Integer(27));
24. pieDataset.setValue("Valeur2", new Integer(10));
25. pieDataset.setValue("Valeur3", new Integer(50));
26. pieDataset.setValue("Valeur4", new Integer(5));
27.
28. JFreeChart pieChart = ChartFactory.createPieChart("Test camembert",
29.     pieDataset, true, true, true);
30. ChartPanel cPanel = new ChartPanel(pieChart);
31. pnl.add(cPanel);
32. }
33.
34. public static void main(String args[]) {
35.     TestPieChart tpc = new TestPieChart();
36.     tpc.setVisible(true);
37. }
38. }

```



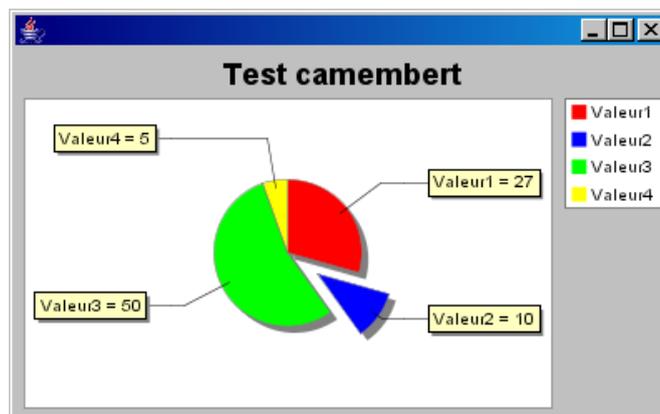
Pour chaque graphique, il existe de nombreuses possibilités de configuration.

Exemple :

```

01. ...
02.     JFreeChart pieChart = ChartFactory.createPieChart("Test camembert",
03.         pieDataset, true, true, true);
04.     PiePlot piePlot = (PiePlot) pieChart.getPlot();
05.     piePlot.setExplodePercent(1, 0.5);
06.     Legend legend = pieChart.getLegend();
07.     legend.setAnchor(Legend.EAST_NORTHEAST);
08.     ChartPanel cPanel = new ChartPanel(pieChart);
09. ...

```



Il est très facile d'exporter le graphique dans un flux.

Exemple : enregistrement du graphique dans un fichier

```

1. ...
2.     File fichier = new File("image.png");
3.     try {
4.         ChartUtilities.saveChartAsPNG(fichier, pieChart, 400, 250);
5.     } catch (IOException e) {

```

```

6.     e.printStackTrace();
7.   }
8.   ...

```

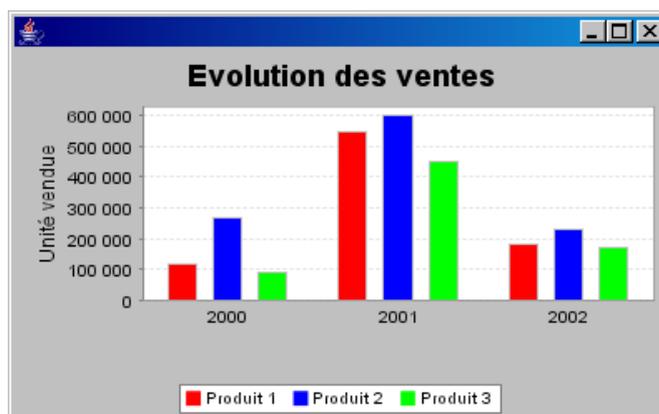
JFreeChart propose aussi plusieurs autres types de graphiques dont les histogrammes.

Exemple : un graphique sous formes de barres

```

01. import java.awt.*;
02. import java.awt.event.*;
03. import javax.swing.*;
04. import org.jfree.chart.*;
05. import org.jfree.chart.plot.*;
06. import org.jfree.data.*;
07.
08. public class TestBarChart extends JFrame {
09.     private JPanel pnl;
10.
11.     public TestBarChart() {
12.         addWindowListener(new WindowAdapter() {
13.             public void windowClosing(WindowEvent e) {
14.                 dispose();
15.                 System.exit(0);
16.             }
17.         });
18.         pnl = new JPanel(new BorderLayout());
19.         setContentPane(pnl);
20.         setSize(400, 250);
21.
22.         DefaultCategoryDataset dataset = new DefaultCategoryDataset();
23.         dataset.addValue(120000.0, "Produit 1", "2000");
24.         dataset.addValue(550000.0, "Produit 1", "2001");
25.         dataset.addValue(180000.0, "Produit 1", "2002");
26.         dataset.addValue(270000.0, "Produit 2", "2000");
27.         dataset.addValue(600000.0, "Produit 2", "2001");
28.         dataset.addValue(230000.0, "Produit 2", "2002");
29.         dataset.addValue(90000.0, "Produit 3", "2000");
30.         dataset.addValue(450000.0, "Produit 3", "2001");
31.         dataset.addValue(170000.0, "Produit 3", "2002");
32.
33.         JFreeChart barChart = ChartFactory.createBarChart("Evolution des ventes", "",
34.             "Unité vendue", dataset, PlotOrientation.VERTICAL, true, true, false);
35.         ChartPanel cPanel = new ChartPanel(barChart);
36.         pnl.add(cPanel);
37.     }
38.
39.     public static void main(String[] args) {
40.         TestBarChart tbc = new TestBarChart();
41.         tbc.setVisible(true);
42.     }
43. }

```



JFreechart peut aussi être mis en oeuvre dans une application web, le plus pratique étant d'utiliser une servlet qui renvoie dans la réponse une image générée par JfreeChart.

Exemple : JSP qui affiche le graphique

```

01. <%@ page language="java" %>
02. <!DOCTYPE HTML PUBLIC "-//w3c//dtd html 4.0 transitional//en">
03. <html>
04. <head>
05. <title>Test JFreeChart</title>
06. </head>
07. <body bgcolor="#FFFFFF">
08. <h1>Exemple de graphique avec JFreeChart</h1>
09. 
10. </body>
11. </html>

```

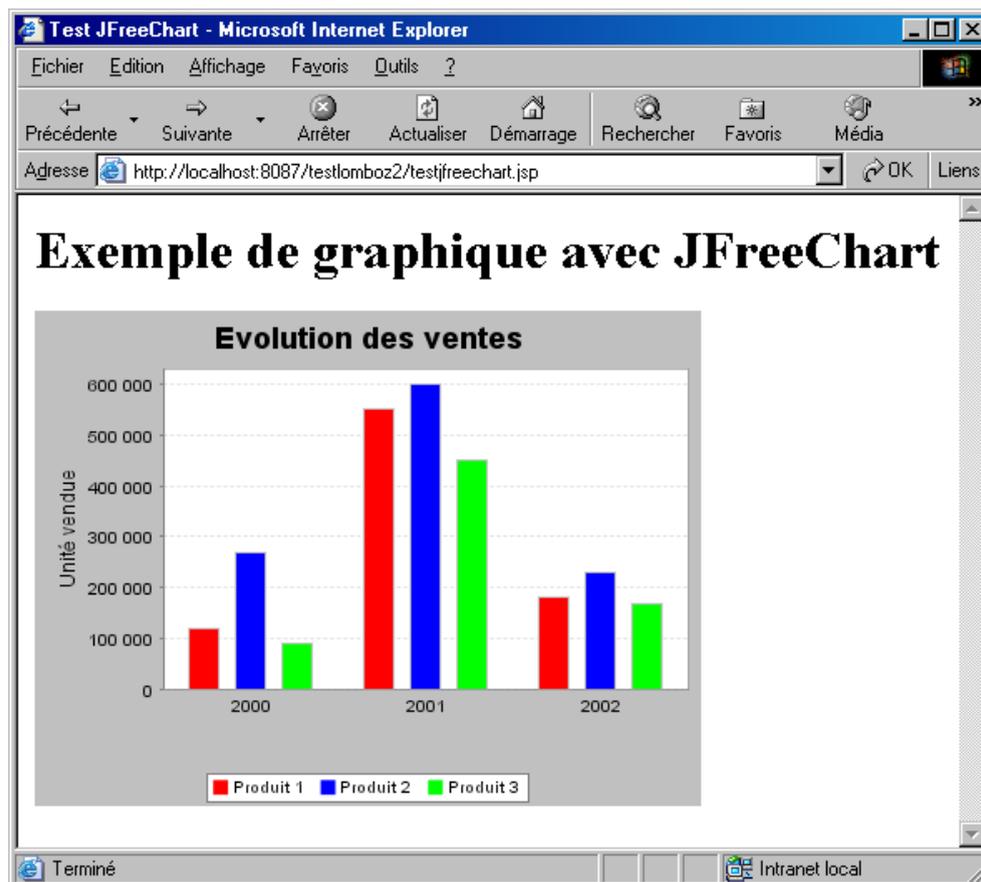
Dans l'exemple précédent, l'image contenant le graphique est générée par une servlet.

Exemple : servlet qui génère l'image

```

01. import java.io.*;
02. import javax.servlet.*;
03. import javax.servlet.http.*;
04. import org.jfree.chart.*;
05. import org.jfree.chart.plot.*;
06. import org.jfree.data.*;
07.
08. public class ServletBarChart extends HttpServlet {
09.
10.     protected void doGet(HttpServletRequest request, HttpServletResponse response)
11.         throws ServletException, IOException {
12.
13.         DefaultCategoryDataset dataset = new DefaultCategoryDataset();
14.         dataset.addValue(120000.0, "Produit 1", "2000");
15.         dataset.addValue(550000.0, "Produit 1", "2001");
16.         dataset.addValue(180000.0, "Produit 1", "2002");
17.         dataset.addValue(270000.0, "Produit 2", "2000");
18.         dataset.addValue(600000.0, "Produit 2", "2001");
19.         dataset.addValue(230000.0, "Produit 2", "2002");
20.         dataset.addValue(90000.0, "Produit 3", "2000");
21.         dataset.addValue(450000.0, "Produit 3", "2001");
22.         dataset.addValue(170000.0, "Produit 3", "2002");
23.
24.         JFreeChart barChart = ChartFactory.createBarChart("Evolution des ventes", "",
25.             "Unité vendue", dataset, PlotOrientation.VERTICAL, true, true, false);
26.         OutputStream out = response.getOutputStream();
27.         response.setContentType("image/png");
28.         ChartUtilities.writeChartAsPNG(out, barChart, 400, 300);
29.     }
30. }

```



Cette section n'a proposé qu'une introduction à JFreeChart en proposant quelques exemples très simples sur les nombreuses possibilités de cette puissante bibliothèque.

98.2. Beanshell



Beanshell est un interpréteur de scripts qu'il est possible d'intégrer dans une application.

Le site officiel du projet est à l'url <http://www.beanshell.org/>

98.3. Apache Commons

Le projet Apache Commons a pour but de fournir des utilitaires

Le site officiel du projet est à l'url <http://commons.apache.org/>.

98.4. Quartz

Quartz est une API qui permet de planifier l'exécution de traitements (jobs scheduler).

Le site officiel du projet est à l'url : <http://quartz-scheduler.org/>.

98.5. JGoodies

Cette bibliothèque propose de faciliter le développement d'applications graphiques utilisant Swing.

Le site officiel du projet est à l'url : <http://www.jgoodies.com/>.

98.6. Apache Lucene

Apache Lucene est un moteur d'indexation et de recherche de texte.

Le site officiel du projet est à l'url : <http://lucene.apache.org/>.

