

```

2  /*
3  * To change this template, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7
8  import java.io.BufferedReader;
9  import java.io.BufferedWriter;
10 import java.io.File;
11 import java.io.FileReader;
12 import java.io.FileWriter;
13 import java.io.IOException;
14 import java.util.StringTokenizer;
15
16 /**
17  *
18  * @author AnneT
19  */
20 public class Fichier {
21
22
23     private    BufferedWriter fW;
24     private    BufferedReader fR;
25     private    String urlFichier;
26
27     public void setUrlFichier(String tmp){
28         urlFichier = tmp;
29     }
30     // Ouvrir un fichier en lecture
31     public boolean ouvrirEnLecture(String nf) {
32         try {
33             setUrlFichier(nf);
34             File f = new File(urlFichier);
35             //(f.exists());
36             fR = new BufferedReader(new FileReader(f));
37             fW = null;
38             // return true;
39             return f.exists();
40         }
41         catch (IOException e) {
42             // System.out.println(urlFichier+" : Erreur à l'ouverture en lecture");
43             return false;
44         }
45     }
46
47     //lire une ligne du fichier
48     public String lire() {
49         try {
50
51             String chaine = fR.readLine();
52             return chaine;
53         }
54         catch (IOException err) {
55
56             return null;
57         }
58     }
59
60     public String [] extraireDonnees(String tmp) {
61         if (tmp != null) {
62             StringTokenizer st = new StringTokenizer(tmp,"#");
63             int i=0;
64             String mot[] = new String[st.countTokens()];
65             while (st.hasMoreTokens()) {
66                 mot[i] = st.nextToken();
67                 i++;
68             }
69             return mot;
70         }
71         else return null;
72     }
73 }
74
75
76 public boolean fermer() {
77     try {
78         if (fR != null) fR.close();
79         else if (fW != null) fW.close();
80         return true;
81     }
82     catch (IOException err) {
83         return false;
84     }
85 }
86 // Ouvrir un fichier en écriture
87 public boolean ouvrirEnEcriture(String nf) {
88     try {
89         setUrlFichier(nf);
90         File f = new File(urlFichier);
91         fW = new BufferedWriter(new FileWriter(f));
92         fR = null;
93         return true;
94     }
95     catch (IOException e) {
96         return false;
97     }
98 }
99
100 // la methode ecrire écrit une chaine et elle retourne à la ligne
101 public boolean ecrire(String tmp) {
102     try {
103         if (tmp != null) {
104             fW.write(tmp,0,tmp.length());
105             fW.newLine();
106         }
107         return true;
108     }
109     catch (IOException err) {
110         return false;
111     }
112 }

```



113 }
114
115
116



Edited with **Infix PDF Editor**
- free for non-commercial use.

To remove this notice, visit
: www.pdfediting.com