

```

1  /*
2   * Serviços de acesso ao dicionário
3   */
4
5  package frases.Services;
6
7  import frases.Core.DomainObjects.*;
8  import frases.Core.Services.IDicionarioService;
9  import frases.DomainObjects.*;
10 import java.io.*;
11
12 /**
13  * Classes com os métodos de acesso ao dicionário
14  * @author Charles Fortes
15  */
16 public class DicionarioService implements IDicionarioService{
17     private static String ERRO_LINHA_INVALIDA_ARQUIVO_DADOS =
18         "Falha ao carregar dados do dicionário.\n"
19         + "Entrada inválida no dicionário:\n";
20
21     private IDicionario _dicionario;
22
23     // <editor-fold defaultstate="collapsed" desc="Constructor">
24     /**
25      * Construtor do Serviço
26      * @throws Exception
27      */
28     public DicionarioService() throws Exception {
29         this.load();
30     }// </editor-fold>
31
32     // <editor-fold defaultstate="collapsed" desc="Public Methods">
33     public IDicionario dicionario() {
34         return _dicionario;
35     }
36
37     public IVerbetes localizar(String palavra) {
38         IVerbetes palavraLocalizada = null;
39         for (IVerbetes p : this._dicionario.verbetes()) {
40             if (p != null && p.palavra().toUpperCase().intern()
41                 == palavra.toUpperCase().intern()) {
42                 palavraLocalizada = p;
43                 break;
44             }
45         }
46         return palavraLocalizada;
47     }
48     // </editor-fold>
49
50     // <editor-fold defaultstate="collapsed" desc="Private Methods">
51
52     private void load() throws Exception
53     {
54         _dicionario = new Dicionario();
55
56         InputStream in = new FileInputStream(CAMINHO_ARQUIVO);
57         InputStreamReader reader = new InputStreamReader(in);
58         BufferedReader data = new BufferedReader(reader);
59
60         while (data.ready())
61         {
62             _dicionario.adicionar(this.Parse(data.readLine()));
63         }

```

```

64     }
65
66     private IVerbete Parse(String dicInputLine) throws Exception
67     {
68         IVerbete verbete = null;
69         if (dicInputLine.charAt(0) == 'V' || dicInputLine.charAt(0) == 'v')
70         {
71             verbete = this.ParseVerbo(dicInputLine);
72         }
73         else if (dicInputLine.charAt(0) == 'S' || dicInputLine.charAt(0) == 's')
74         {
75             verbete = this.ParseSubstantivo(dicInputLine);
76         }
77         else
78         {
79             throw new Exception(ERRO_LINHA_INVALIDA_ARQUIVO_DADOS
80                 + "\"" + dicInputLine + "\"");
81         }
82         return verbete;
83     }
84
85     private IVerbete ParseVerbo(String verboString) throws Exception
86     {
87         String[] dadosLinha = verboString.split(";");
88         if (dadosLinha.length != 5 || (!dadosLinha[4].equalsIgnoreCase("I") &&
89             !dadosLinha[4].equalsIgnoreCase("TD") ))
90             throw new Exception(ERRO_LINHA_INVALIDA_ARQUIVO_DADOS
91                 + "\"" + verboString + "\"");
92
93         TipoVerbo tpVerbo = (dadosLinha[4].equalsIgnoreCase("I")) ?
94             TipoVerbo.INTRANSITIVO : TipoVerbo.TRANSITIVO_DIRETO;
95
96         return new Verbo(dadosLinha[1], dadosLinha[2], dadosLinha[3], tpVerbo);
97     }
98
99     private IVerbete ParseSubstantivo(String substantivoString) throws Exception
100    {
101        String[] dadosLinha = substantivoString.split(";");
102
103        if (dadosLinha.length != 4 || (!dadosLinha[2].equalsIgnoreCase("M") &&
104            !dadosLinha[2].equalsIgnoreCase("F") )
105            || (!dadosLinha[3].equalsIgnoreCase("S") &&
106            !dadosLinha[3].equalsIgnoreCase("P") ))
107            throw new Exception(ERRO_LINHA_INVALIDA_ARQUIVO_DADOS
108                + "\"" + substantivoString + "\"");
109
110        GeneroSubstantivo genero = (dadosLinha[2].equalsIgnoreCase("M")) ?
111            GeneroSubstantivo.MASCULINO : GeneroSubstantivo.FEMININO;
112
113        NumeroSubstantivo numero = (dadosLinha[3].equalsIgnoreCase("S")) ?
114            NumeroSubstantivo.SINGULAR : NumeroSubstantivo.PLURAL;
115
116        return new Substantivo(dadosLinha[1], genero, numero);
117    }
118    // </editor-fold>
119
120 }

```