

Oct 12, 05 13:14

NameDataRetriever.java

Page 1/3

```

import java.net.*;
import java.util.*;
import java.io.*;
5
import javax.swing.text.MutableAttributeSet;
import javax.swing.text.html.HTML;
import javax.swing.text.html.HTMLEditorKit;

10
public class NameDataRetriever {

    private URL myURL;
    private URLConnection myConnection;
15
    private DataOutputStream myOutput;

    /**
     * Open a URL (to census/name/social-security data site)
     * and populate private state so that information for
     * a specific year and number of names can be read. Call open
     * before calling getInfo.
     * @param url is the URL for ss/gov data
     * @return true if url opened successfully, false otherwise
     */
25
    public boolean open(String url) {
        try {
            myURL = new URL(url);
            myConnection = myURL.openConnection();

30
            myConnection.setDoInput(true);
            myConnection.setDoOutput(true);
            myConnection.setUseCaches(false);
            myConnection.setRequestProperty(
35
                "Content-Type",
                "application/x-www-form-urlencoded");
            myOutput = new DataOutputStream(myConnection.getOutputStream());

            return true;
        }
        catch (MalformedURLException e) {
40
            return false;
        }
        catch (IOException e) {
            return false;
        }
45
    }

    /**
     * Get information for a specific year from the previously opened
     * URL and print it to standard out.
     * @param year is year for which information requested
     * @param rank is the number of names requested, e.g., 100 for top 100 n
50
     *ames
     */
    public void getInfo(String year, String rank) {
60
        String params = "year="+URLEncoder.encode(year);
        String params2 = "top="+URLEncoder.encode(rank);
        params = params+"&"+params2;
        try {
            myOutput.writeBytes(params);
            myOutput.close();

            LocalParser local = new LocalParser();
            HTMLEditorKit.Parser parser = local.getParser();
            BufferedReader reader = new BufferedReader(new InputStreamReader(my
            Connection.getInputStream()));
65
            parser.parse(reader, new LocalCallback(), true);

        } catch (IOException e) {
            System.out.println("error reading data");
        }
70
    }

```

Oct 12, 05 13:14

NameDataRetriever.java

Page 2/3

```

    public static void main(String[] args) {
        NameDataRetriever p = new NameDataRetriever();
        String url = "http://www.ssa.gov/cgi-bin/popularnames.cgi";
        p.open(url);
        p.getInfo("2004", "100");
    }
75

    /**
     * Ignore everything below here
     */
    private class LocalParser extends HTMLEditorKit {
80
        public HTMLEditorKit.Parser getParser() {
            return super.getParser();
        }
    }

    /**
     * Inner class for processing all "<a href..>" tags when
     * reading a base URL. This callback stores the relative
     * URLs in an ArrayList of the class it's nested in.
     * @author ola
     */
95
    private class LocalCallback extends HTMLEditorKit.ParserCallback {

        private boolean myTagStart = false;
        private boolean myRankStart = false;
        private int myRank = 1;
        private int myTDcount = 0;
        public void handleStartTag(HTML.Tag t, MutableAttributeSet a, int pos
100
        s) {
            if (t == HTML.Tag.TD) {
                myTagStart = true;
            }
            else {
                myTagStart = false;
            }
105
        }
        public void handleEndTag(HTML.Tag t, int pos) {
            if (t == HTML.Tag.TD) {
                if (myRankStart) {
                    myTDcount++;
                }
            }
            myTagStart = false;
110
        }
        public void handleText(char[] data, int pos) {
            String s = new String(data);
            if (myTagStart) {
                int x = -1;
                try {
                    x = Integer.parseInt(s);
                    if (x == 1) {
                        myRankStart = true;
                    }
                }
                catch (NumberFormatException fe) {
                    if (myRankStart) {
                        if (myTDcount % 3 == 1) {
                            System.out.print(myRank+
115
                            "\t"+s+"\t");
                        }
                        else {
                            System.out.println(s);
                            myRank++;
                        }
                    }
                }
            }
            myTagStart = false;
120
        }
    }
}
135
140

```

```
}  
}
```