

```

1 // foj/10/juno/ShellCommandTable.java
2 //
3 //
4 // Copyright 2003 Bill Campbell and Ethan Bolker
5
6 import java.util.*;
7
8 /**
9  * A ShellCommandTable object maintains a dispatch table of
10 * ShellCommand objects keyed by the command names used to invoke
11 * them.
12 *
13 * To add a new shell command to the table, install it from
14 * method fillTable().
15 *
16 * @see ShellCommand
17 *
18 * @version 10
19 */
20
21 public class ShellCommandTable
22 implements java.io.Serializable
23 {
24     private Map table = new TreeMap();
25
26     /**
27      * Construct and fill a shell command table.
28      */
29
30     public ShellCommandTable()
31     {
32         fillTable();
33     }
34
35     /**
36      * Get a ShellCommand, given the command name key.
37      *
38      * @param key the name associated with the command we're
39      *         looking for.
40      *
41      * @return the command we're looking for, null if none.
42      */
43
44     public ShellCommand lookup( String key )
45     {
46         ShellCommand commandObject = (ShellCommand) table.get( key );
47         if (commandObject != null) {
48             return commandObject;
49         }
50
51         // try to load dynamically
52         // construct classname = "KeyCommand"
53         char[] chars = (key + "Command").toCharArray();
54         chars[0] = key.toUpperCase().charAt(0);
55         String classname = new String(chars);
56         try {

```

```

57         commandObject =
58             (ShellCommand)Class.forName(classname).newInstance();
59     }
60     catch (Exception e) { // couldn't find class
61         return null;
62     }
63     install(key, commandObject); // put it in table for next time
64     return commandObject;
65 }
66
67 /**
68  * Get an array of the command names.
69  *
70  * @return the array of command names.
71  */
72
73     public String[] getCommandNames()
74     {
75         return (String[]) table.keySet().toArray( new String[0] );
76     }
77
78     // Associate a command name with a ShellCommand.
79
80     private void install( String commandName, ShellCommand command )
81     {
82         table.put( commandName, command );
83     }
84
85     // Fill the dispatch table with ShellCommands, keyed by their
86     // command names.
87
88     private void fillTable()
89     {
90         install( "list", new ListCommand() );
91         install( "cd", new CdCommand() );
92         install( "newfile", new NewfileCommand() );
93         install( "remove", new RemoveCommand() );
94         install( "help", new HelpCommand() );
95         install( "mkdir", new MkdirCommand() );
96         install( "type", new TypeCommand() );
97         install( "logout", new LogoutCommand() );
98     }
99 }

```