```
 1  // joi/7/juno/ShellCommand.java
 2  //
 3  //
 4  // Copyright 2003 Bill Campbell and Ethan Bolker
 5
 6  import java.util.*;
 7
 8  /**
 9   * Model those features common to all ShellCommands.
10   *
11   * Each concrete extension of this class provides a constructor
12   * and an implementation for method doIt.
13   *
14   * @version 7
15   */
16
17  public abstract class ShellCommand
18  {
19      private String helpString;   // documents the command
20      private String argString;    // any args to the command
21
22      /**
23       * A constructor, always called (as super()) by the subclass.
24       * Used only for commands that have arguments.
25       *
26       * @param helpString a brief description of what the command does.
27       * @param argString a prototype illustrating the required arguments.
28       */
29
30      protected ShellCommand( String helpString, String argString )
31      {
32          this.argString  = argString;
33          this.helpString = helpString;
34      }
35
36      /**
37       * A constructor for commands having no arguments.
38       *
39       * @param helpString a brief description of what the command does.
40       */
41
42      protected ShellCommand( String helpString )
43      {
44          this( helpString, "" );
45      }
46
47      /**
48       * Execute the command.
49       *
50       * @param args the remainder of the command line.
51       * @param sh   the current shell
52       *
53       * @exception JunoException for reporting errors
54       */
55
56      public abstract void doIt( StringTokenizer args, Shell sh )
```



```
57          throws JunoException;
58
59      /**
60       * Help for this command.
61       *
62       * @return the help string.
63       */
64
65      public String getHelpString()
66      {
67          return helpString;
68      }
69
70      /**
71       * The argument string prototype.
72       *
73       * @return the argument string prototype.
74       */
75
76      public String getArgString()
77      {
78          return argString;
79      }
80  }
```