

```

1 // fo1/5/shapes/Line.java
2 //
3 //
4 // Copyright 2003 Bill Campbell and Ethan Bolker
5
6 /**
7  * A Line has a length and a paintChar used to paint
8  * itself on a Screen.
9  *
10 * Subclasses of this abstract class specify the direction
11 * of the line.
12 *
13 * @version 5
14 */
15
16 public abstract class Line
17 {
18     protected int length; // length in (character) pixels.
19     protected char paintChar; // character used for painting.
20
21     /**
22      * Construct a Line.
23      *
24      * @param length length in (character) pixels.
25      * @param paintChar character used for painting this Line.
26      */
27     protected Line( int length, char paintChar )
28     {
29         this.length = length;
30         this.paintChar = paintChar;
31     }
32
33     /**
34      * Get the length of this line.
35      *
36      * @return the length in (character) pixels.
37      */
38     public int getLength()
39     {
40         return length;
41     }
42
43     /**
44      * Set the length of this line.
45      *
46      * @param length the new length in (character) pixels.
47      */
48     public void setLength( int length )
49     {
50         this.length = length;
51     }
52
53     /**
54      */
55
56

```

```

57     * Get the paintChar of this Line.
58     *
59     * @return the paintChar.
60     */
61
62     public char getPaintChar()
63     {
64         return paintChar;
65     }
66
67     /**
68      * Set the paintChar of this Line.
69      *
70      * @param paintChar the new paintChar.
71      */
72
73     public void setPaintChar( char paintChar )
74     {
75         this.paintChar = paintChar;
76     }
77
78     /**
79      * Paint this Line on Screen s at position (x,y).
80      *
81      * @param s the Screen on which this Line is to be painted.
82      * @param x the x position for the line.
83      * @param y the y position for the line.
84      */
85
86     public abstract void paintOn( Screen s, int x, int y );
87
88     /**
89      * Paint this Line on Screen s at position (0,0).
90      *
91      * @param s the Screen on which this Line is to be painted.
92      */
93
94     public void paintOn( Screen s )
95     {
96         paintOn( s, 0, 0 );
97     }
98

```