

```

1 package maze.display;
2
3 import java.awt.event.KeyEvent;
10
11 /**
12  * Links key presses and actions
13  *
14  * @author Pierre-Andre Mudry
15  * @date February 2012
16  * @version 1.0
17  */
18 public class KeyboardListener implements KeyListener {
19
20     MazeGame mg;
21     MazeContainer mc;
22
23     /**
24      * To link keys to actions from the game in the maze, we need references to
25      * both
26      *
27      * @param mc The maze
28      * @param mg The game
29      */
30     public KeyboardListener(MazeContainer mc, MazeGame mg) {
31         this.mc = mc;
32         this.mg = mg;
33     }
34
35     /**
36      * What happens when a key has been pressed
37      */
38     @Override
39     public void keyPressed(KeyEvent arg0) {
40
41         /**
42          * Keys for player 1
43          */
44         switch (arg0.getKeyCode()) {
45             case KeyEvent.VK_W:
46                 mg.movePlayer(Direction.UP);
47                 break;
48             case KeyEvent.VK_S:
49                 mg.movePlayer(Direction.DOWN);
50                 break;
51             case KeyEvent.VK_D:
52                 mg.movePlayer(Direction.RIGHT);
53                 break;
54             case KeyEvent.VK_A:
55                 mg.movePlayer(Direction.LEFT);
56                 break;
57             case KeyEvent.VK_Q:
58                 mg.displaySolution();
59                 break;
60             case KeyEvent.VK_N:
61                 mg.generateNewMaze(new Random().nextInt());
62                 break;
63         }
64     }
65
66     /**
67      * This method is called when a key has been released (i.e. no more pressed)
68      */
69     @Override
70     public void keyReleased(KeyEvent arg0) {
71
72     }
73
74     /**
75      * This method is called when a key has been pressed and released (complete
76      * cycle)
77      */
78     @Override
79     public void keyTyped(KeyEvent arg0) {
80
81     }
82 }
83

```