

```
1 import components.stack.Stack;
2
3 /**
4  * Controller class.
5  *
6  * @author Bruce W. Weide
7  * @author Paolo Bucci
8  */
9 public final class AppendUndoController1 implements
AppendUndoController {
10
11     /**
12      * Model object.
13      */
14     private final AppendUndoModel model;
15
16     /**
17      * View object.
18      */
19     private final AppendUndoView view;
20
21     /**
22      * Stack to get the last string.
23      */
24     private final Stack<String> sentence;
25
26     /**
27      * Updates view to display model.
28      *
29      * @param model
30      *         the model
31      * @param view
32      *         the view
33      */
34     private static void updateViewToMatchModel(AppendUndoModel
model,
35         AppendUndoView view) {
36         /**
37          * Get model info
38          */
39         String input = model.input();
40         Stack<String> output = model.output();
41         /**
42          * Update view to reflect changes in model
```

```
43         */
44         view.updateInputDisplay(input);
45         view.updateOutputDisplay(output.pop());
46     }
47
48     /**
49      * Constructor; connects {@code this} to the model and view it
      coordinates.
50      *
51      * @param model
52      *         model to connect to
53      * @param view
54      *         view to connect to
55      */
56     public AppendUndoController1(AppendUndoModel model,
      AppendUndoView view) {
57         this.model = model;
58         this.view = view;
59         /*
60          * Update view to reflect initial value of model
61          */
62         updateViewToMatchModel(this.model, this.view);
63     }
64
65     /**
66      * Processes reset event.
67      */
68     @Override
69     public void processResetEvent() {
70         /*
71          * Update model in response to this event
72          */
73         this.model.setInput("");
74         this.model.setOutput("");
75         //Couldn't figure out how to do it with Stacks
76         /*
77          * Update view to reflect changes in model
78          */
79         updateViewToMatchModel(this.model, this.view);
80     }
81
82     /**
83      * Processes copy event.
84      */
```

```
85     * @param input
86     *         value of input text (provided by view)
87     */
88     @Override
89     public void processCopyEvent(String input) {
90         /*
91          * Update model in response to this event
92          */
93         this.model.setInput(input);
94         this.model.setOutput(input);
95         this.sentence.clear();
96         this.sentence.push(input);
97         //Couldn't figure out how to do it with Stacks
98
99         /*
100          * Update view to reflect changes in model
101          */
102         updateViewToMatchModel(this.model, this.view);
103     }
104
105     @Override
106     public void processAppendEvent(String input) {
107         String result = this.model.input() + input;
108         this.model.setOutput(result);
109
110         this.sentence.push(input);
111
112         updateViewToMatchModel(this.model, this.view);
113     }
114
115     @Override
116     public void processUndoEvent() {
117         this.sentence.flip();
118         String popped = this.sentence.pop();
119         this.sentence.flip();
120         String result = "";
121         for (String e : this.sentence) {
122             result += e;
123         }
124         this.model.setInput(popped);
125         this.model.setOutput(result);
126
127     }
128
```

AppendUndoController1.java

Thursday, December 2, 2021, 10:19 PM

129 }

130