

# NNCalcController1.java

```

1 import components.naturalnumber.NaturalNumber;
2
3
4 /**
5  * Controller class.
6  *
7  * @author Put your name here
8  */
9 public final class NNCalcController1 implements NNCalcController {
10
11     /**
12      * Model object.
13      */
14     private final NNCalcModel model;
15
16     /**
17      * View object.
18      */
19     private final NNCalcView view;
20
21     /**
22      * Useful constants.
23      */
24     private static final NaturalNumber TWO = new NaturalNumber2(2),
25         INT_LIMIT = new NaturalNumber2(Integer.MAX_VALUE);
26
27     /**
28      * Updates this.view to display this.model, and to allow only operations
29      * that are legal given this.model.
30      *
31      * @param model
32      *         the model
33      * @param view
34      *         the view
35      * @ensures [view has been updated to be consistent with model]
36      */
37     private static void updateViewToMatchModel(NNCalcModel model,
38         NNCalcView view) {
39
40         // TODO: fill in body
41
42     }
43
44     /**
45      * Constructor.
46      *
47      * @param model
48      *         model to connect to
49      * @param view
50      *         view to connect to
51      */
52     public NNCalcController1(NNCalcModel model, NNCalcView view) {
53         this.model = model;
54         this.view = view;
55         updateViewToMatchModel(model, view);
56     }
57
58     @Override

```

```

59     public void processClearEvent() {
60         /*
61          * Get alias to bottom from model
62          */
63         NaturalNumber bottom = this.model.bottom();
64         /*
65          * Update model in response to this event
66          */
67         bottom.clear();
68         /*
69          * Update view to reflect changes in model
70          */
71         updateViewToMatchModel(this.model, this.view);
72     }
73
74     @Override
75     public void processSwapEvent() {
76         /*
77          * Get aliases to top and bottom from model
78          */
79         NaturalNumber top = this.model.top();
80         NaturalNumber bottom = this.model.bottom();
81         /*
82          * Update model in response to this event
83          */
84         NaturalNumber temp = top.newInstance();
85         temp.transferFrom(top);
86         top.transferFrom(bottom);
87         bottom.transferFrom(temp);
88         /*
89          * Update view to reflect changes in model
90          */
91         updateViewToMatchModel(this.model, this.view);
92     }
93
94     @Override
95     public void processEnterEvent() {
96
97         // TODO: fill in body
98
99     }
100
101     @Override
102     public void processAddEvent() {
103
104         // TODO: fill in body
105
106     }
107
108     @Override
109     public void processSubtractEvent() {
110
111         // TODO: fill in body
112
113     }
114
115     @Override

```

NNCalcController1.java

```
116     public void processMultiplyEvent() {
117
118         // TODO: fill in body
119
120     }
121
122     @Override
123     public void processDivideEvent() {
124
125         // TODO: fill in body
126
127     }
128
129     @Override
130     public void processPowerEvent() {
131
132         // TODO: fill in body
133
134     }
135
136     @Override
137     public void processRootEvent() {
138
139         // TODO: fill in body
140
141     }
142
143     @Override
144     public void processAddNewDigitEvent(int digit) {
145
146         // TODO: fill in body
147
148     }
149 }
150 }
151
```