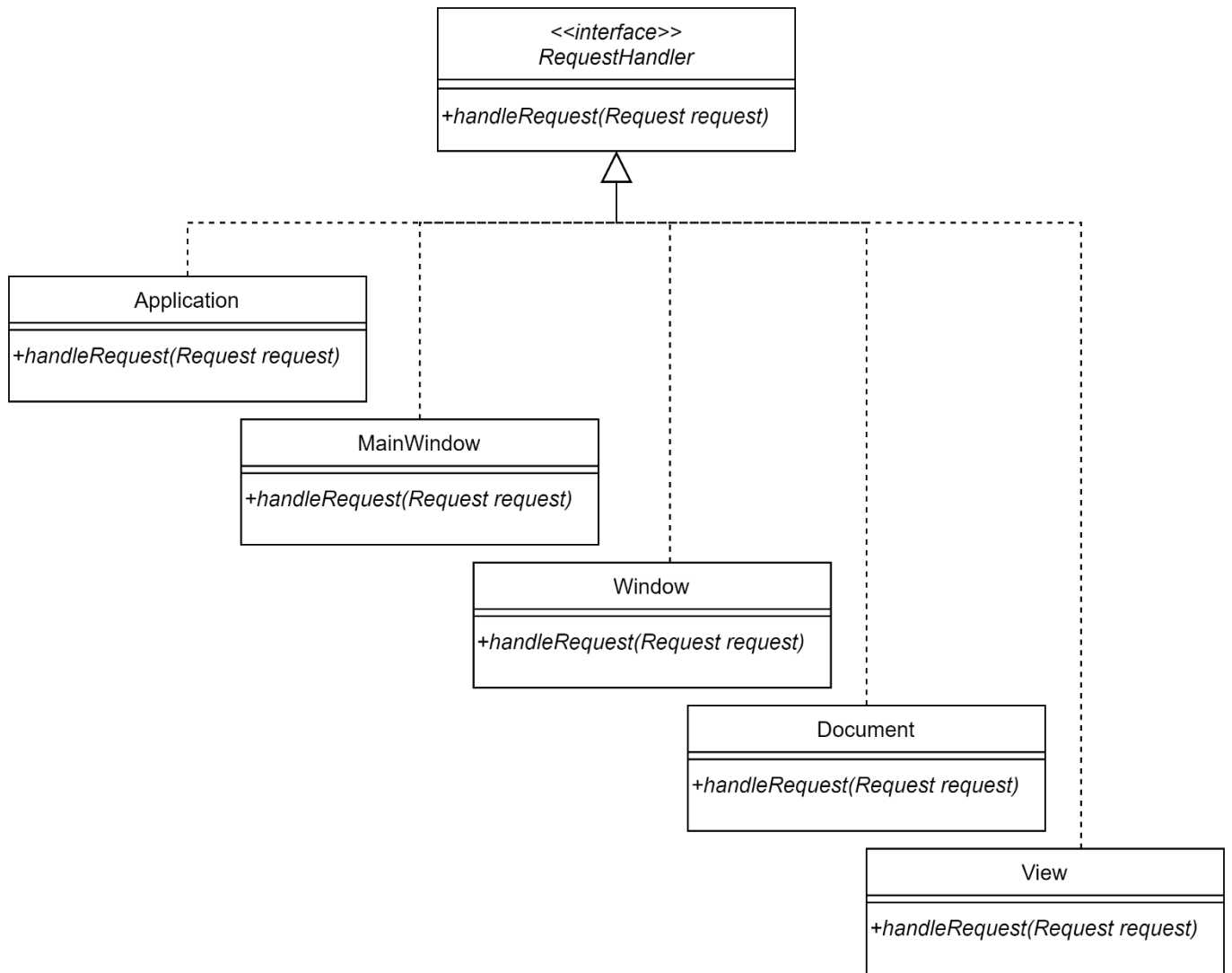


1) 개요



2) Example1.java

```
1 package chain_of_resposibility;
2
3 import java.util.Scanner;
4
5 enum Request {
6     CLOSE, COPY, CUT, EXIT, DELETE, LOAD_RESOURCE, OPEN_FILE, PASTE, SAVE_FILE, TILE_VERTICALLY,
7     TILE_HORIZONTALLY, UNDO, ZOOM
8 }
9
10 interface RequestHandler {
11     void handleRequest(Request req);
12 }
13
14 class Application implements RequestHandler {
15     @Override
16     public void handleRequest(Request req) {
17         switch (req) {
18             case EXIT:
19                 System.out.printf("Application - ExitWn");
20                 break;
21             case LOAD_RESOURCE:
22                 System.out.printf("Application - LoadResourceWn");
23                 break;
24             default:
25                 System.out.printf("Application - Unknown RequestWn");
26                 break;
27         }
28     }
29 }
30
31 class MainWindow implements RequestHandler {
32     Application app;
33
34     public MainWindow(Application a) {
35         app = a;
36     }
37
38     @Override
39     public void handleRequest(Request req) {
40         switch (req) {
41             case OPEN_FILE:
42                 System.out.printf("MainWindow - Open FileWn");
43                 break;
44             case TILE_VERTICALLY:
45                 System.out.printf("MainWindow - Tile VerticallyWn");
46                 break;
47             case TILE_HORIZONTALLY:
48                 System.out.printf("MainWindow - Tile HorizontallyWn");
49                 break;
50             default:
51                 app.handleRequest(req);
52                 break;
53         }
54     }
55 }
56
57 class Window implements RequestHandler {
58     MainWindow mainWnd;
59
60     public Window(MainWindow m) {
61         mainWnd = m;
62     }
63
64     @Override
65     public void handleRequest(Request req) {
66         switch (req) {
67             case CLOSE:
68                 System.out.printf("Window - CloseWn");
```

```

69         break;
70     default:
71         mainWnd.handleRequest(req);
72         break;
73     }
74 }
75 }
76
77 class Document implements RequestHandler {
78     Window win;
79
80     public Document(Window w) {
81         win = w;
82     }
83
84     @Override
85     public void handleRequest(Request req) {
86         switch (req) {
87             case SAVE_FILE:
88                 System.out.printf("Document1 - Save FileWn");
89                 break;
90             case UNDO:
91                 System.out.printf("Document1 - UndoWn");
92                 break;
93             default:
94                 win.handleRequest(req);
95                 break;
96         }
97     }
98 }
99
100 class View implements RequestHandler {
101     Document doc;
102     Window win;
103     static View topView;
104
105     public View(Document d) {
106         doc = d;
107     }
108
109     public void activate() {
110         topView = this;
111     }
112
113     public static View getTopView() {
114         return topView;
115     }
116
117     @Override
118     public void handleRequest(Request req) {
119         switch (req) {
120             case ZOOM:
121                 System.out.printf("View - ZoomWn");
122                 break;
123             default:
124                 doc.handleRequest(req);
125                 break;
126         }
127     }
128 }
129
130 class View1 extends View {
131     public View1(Document d) {
132         super(d);
133     }
134
135     @Override
136     public void handleRequest(Request req) {
137         switch (req) {

```

```

138     case COPY:
139         System.out.printf("View1 - CopyWn");
140         break;
141     case CUT:
142         System.out.printf("View1 - CutWn");
143         break;
144     case PASTE:
145         System.out.printf("View1 - PasteWn");
146         break;
147     case DELETE:
148         System.out.printf("View1 - DeleteWn");
149         break;
150     default:
151         super.handleRequest(req);
152         break;
153     }
154 }
155 }
156
157 class View2 extends View {
158     public View2(Document d) {
159         super(d);
160     }
161
162     @Override
163     public void handleRequest(Request req) {
164         switch (req) {
165             case COPY:
166                 System.out.printf("View2 - CopyWn");
167                 break;
168             case CUT:
169                 System.out.printf("View2 - CutWn");
170                 break;
171             case PASTE:
172                 System.out.printf("View2 - PasteWn");
173                 break;
174             case DELETE:
175                 System.out.printf("View2 - DeleteWn");
176                 break;
177             default:
178                 super.handleRequest(req);
179                 break;
180         }
181     }
182 }
183
184 public class Example1 {
185     static View view1, view2;
186
187     public static void createApplication() {
188         Application app = new Application();
189         MainWindow mainWin = new MainWindow(app);
190         Window win = new Window(mainWin);
191         Document doc = new Document(win);
192         view1 = new View1(doc);
193         view2 = new View2(doc);
194     }
195
196     public static void main(String[] args) {
197         try (Scanner scanner = new Scanner(System.in)) {
198             createApplication();
199             Request[] requests = Request.values();
200             while (true) {
201                 for (int i = 0; i < requests.length; ++i)
202                     System.out.printf("%d) %sWn", i, requests[i]);
203                 System.out.printf("Choice? ");
204                 int index = Integer.parseInt(scanner.next());
205
206                 if (index % 2 == 1)

```

```

207         view1.activate();
208     else
209         view2.activate();
210
211     System.out.println();
212     View.getTopView().handleRequest(requests[index]);
213     System.out.println();
214
215     if (requests[index] == Request.EXIT)
216         break;
217     }
218 }
219 }
220 }
221

```

실행 사례

0) CLOSE
 1) COPY
 2) CUT
 3) EXIT
 4) DELETE
 5) LOAD_RESOURCE
 6) OPEN_FILE
 7) PASTE
 8) SAVE_FILE
 9) TILE_VERTICALLY
 10) TILE_HORIZONTALLY
 11) UNDO
 12) ZOOM

Choice? 6

MainWindow - Open File

0) CLOSE
 1) COPY
 2) CUT
 3) EXIT
 4) DELETE
 5) LOAD_RESOURCE
 6) OPEN_FILE
 7) PASTE
 8) SAVE_FILE
 9) TILE_VERTICALLY
 10) TILE_HORIZONTALLY
 11) UNDO
 12) ZOOM

Choice? 8

Document1 - Save File

0) CLOSE
 1) COPY
 2) CUT
 3) EXIT
 4) DELETE
 5) LOAD_RESOURCE
 6) OPEN_FILE
 7) PASTE
 8) SAVE_FILE
 9) TILE_VERTICALLY
 10) TILE_HORIZONTALLY
 11) UNDO
 12) ZOOM

Choice? 2

View2 - Cut

- 0) CLOSE
- 1) COPY
- 2) CUT
- 3) EXIT
- 4) DELETE
- 5) LOAD_RESOURCE
- 6) OPEN_FILE
- 7) PASTE
- 8) SAVE_FILE
- 9) TILE_VERTICALLY
- 10) TILE_HORIZONTALLY
- 11) UNDO
- 12) ZOOM

Choice? 7

View1 - Paste

- 0) CLOSE
- 1) COPY
- 2) CUT
- 3) EXIT
- 4) DELETE
- 5) LOAD_RESOURCE
- 6) OPEN_FILE
- 7) PASTE
- 8) SAVE_FILE
- 9) TILE_VERTICALLY
- 10) TILE_HORIZONTALLY
- 11) UNDO
- 12) ZOOM

Choice? 11

Document1 - Undo

- 0) CLOSE
- 1) COPY
- 2) CUT
- 3) EXIT
- 4) DELETE
- 5) LOAD_RESOURCE
- 6) OPEN_FILE
- 7) PASTE
- 8) SAVE_FILE
- 9) TILE_VERTICALLY
- 10) TILE_HORIZONTALLY
- 11) UNDO
- 12) ZOOM

Choice? 9

MainWindow - Tile Vertically

- 0) CLOSE
- 1) COPY
- 2) CUT
- 3) EXIT
- 4) DELETE
- 5) LOAD_RESOURCE
- 6) OPEN_FILE
- 7) PASTE
- 8) SAVE_FILE
- 9) TILE_VERTICALLY
- 10) TILE_HORIZONTALLY
- 11) UNDO
- 12) ZOOM

Choice?