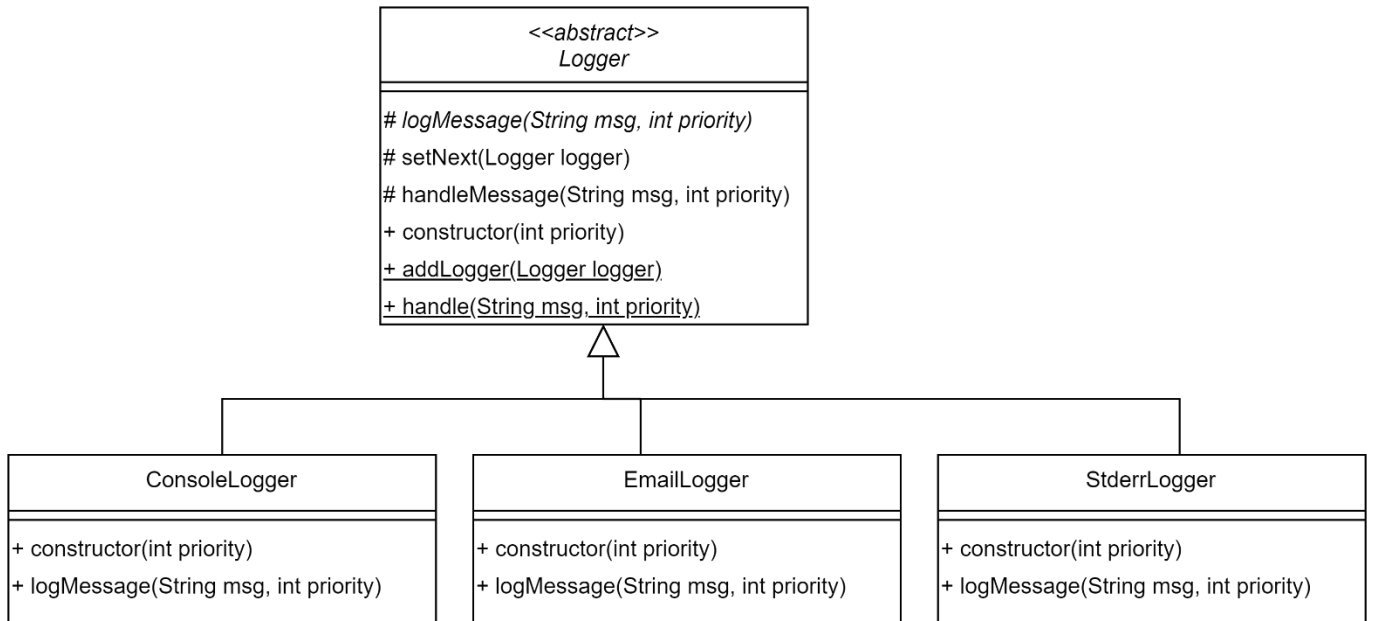


1) 개요



2) Example2.java

```
1 package chain_of_resposibility.e2;
2
3 abstract class Logger {
4
5     public static final int DEBUG = 1;
6     public static final int INFO = 2;
7     public static final int ERROR = 3;
8
9     protected int mask;
10    protected Logger next;
11    protected static Logger head;
12
13    protected void setNext(Logger l) {
14        next = l;
15    }
16
17    protected void handleMessage(String msg, int priority) {
18        if (priority >= mask)
19            logMessage(msg, priority);
20        if (next != null)
21            next.handleMessage(msg, priority);
22    }
23
24    protected abstract void logMessage(String msg, int priority);
25
26    public Logger(int priority) {
27        this.mask = priority;
28    }
29
30    public static void addLogger(Logger logger) {
31        logger.setNext(head);
32        head = logger;
33    };
34
35    public static void handle(String msg, int priority) {
36        head.handleMessage(msg, priority);
37    }
38 }
39
40 class ConsoleLogger extends Logger {
41     public ConsoleLogger(int priority) {
42         super(priority);
43     }
44
45     @Override
46     public void logMessage(String msg, int priority) {
47         System.out.printf("Writing to console: %d, %s\n", priority, msg);
48     }
49 }
50
51 class EmailLogger extends Logger {
52     public EmailLogger(int priority) {
53         super(priority);
54     }
55
56     @Override
57     public void logMessage(String msg, int priority) {
58         System.out.printf("Sending via e-mail: %d, %s\n", priority, msg);
59     }
60 }
61
62 class StderrLogger extends Logger {
63     public StderrLogger(int priority) {
64         super(priority);
65     }
66
67     @Override
68     public void logMessage(String msg, int priority) {
```

```

69         System.out.printf("Writing to stderr: %d, %s\n", priority, msg);
70     }
71 }
72
73 //-----
74
75 public class Example2 {
76
77     static void createLogger() {
78         // ERROR 등급 이상의 메시지를 이메일로 전송
79         Logger.addLogger(new EmailLogger(Logger.ERROR));
80
81         // INFO 등급 이상의 메시지를 stderr 스트림에 출력
82         Logger.addLogger(new StderrLogger(Logger.INFO));
83
84         // DEBUG 등급 이상의 메시지를 console에 출력
85         Logger.addLogger(new ConsoleLogger(Logger.DEBUG));
86     }
87
88     public static void main(String[] args) {
89         createLogger();
90
91         // ERROR 등급의 메시지 출력
92         Logger.handle("An error has occurred.", Logger.ERROR);
93
94         // INFO 등급의 메시지 출력
95         Logger.handle("Step1 completed.", Logger.INFO);
96
97         // DEBUG 등급의 메시지 출력
98         Logger.handle("Entering function y.", Logger.DEBUG);
99     }
100 }
101

```

출력

```

Writing to console: 3, An error has occurred.
Writing to stderr: 3, An error has occurred.
Sending via e-mail: 3, An error has occurred.
Writing to console: 2, Step1 completed.
Writing to stderr: 2, Step1 completed.
Writing to console: 1, Entering function y.

```