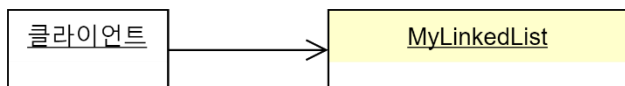
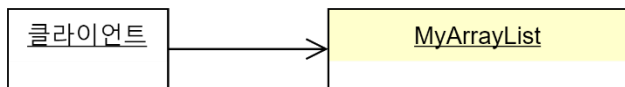
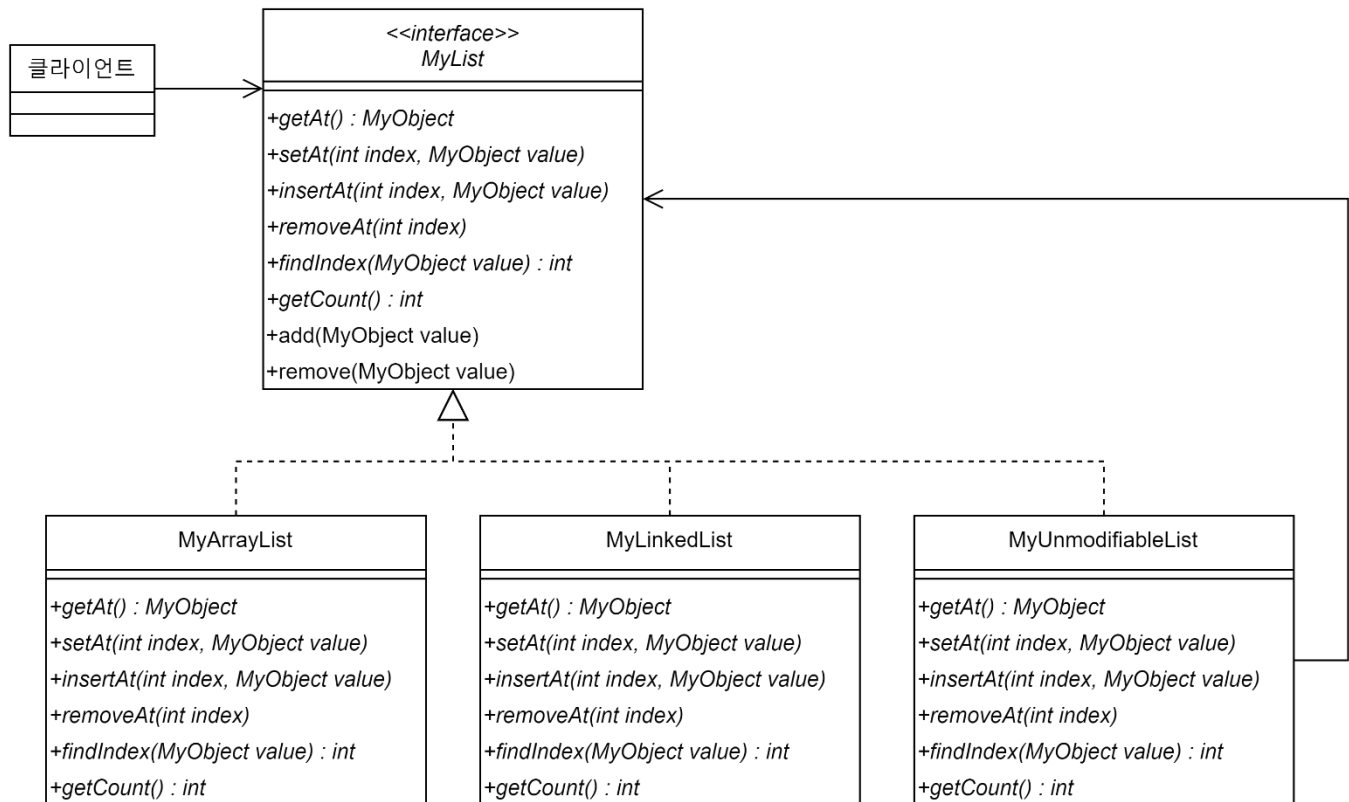
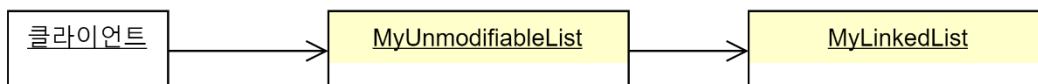
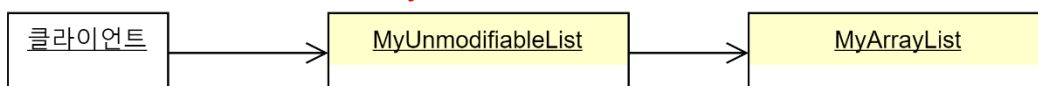


1) 개요



Proxy? Decorator?



2) MyUnmodifiableList.java

```
1 package decorator.u2;
2
3 public class MyUnmodifiableList implements MyList {
4     MyList list;
5
6     MyUnmodifiableList(MyList list) {
7         this.list = list;
8     }
9
10    @Override
11    public int getCount() {
12        return list.getCount();
13    }
14
15    @Override
16    public MyObject getAt(int index) {
17        return list.getAt(index);
18    }
19
20    @Override
21    public void setAt(int index, MyObject data) {
22    }
23
24    @Override
25    public void insertAt(int index, MyObject data) {
26    }
27
28    @Override
29    public void removeAt(int index) {
30    }
31
32    @Override
33    public int findIndex(MyObject data) {
34        return list.findIndex(data);
35    }
36 }
```

3) Example2.java

```
1 package decorator.u2;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class Example2 {
7
8     static class Worker extends Thread {
9         MyList list;
10
11         Worker(MyList list) {
12             this.list = list;
13         }
14
15         @Override
16         public void run() {
17             for (int i = 0; i < 1000; ++i) {
18                 list.insertAt(0, new MyInt(999));
19                 list.removeAt(0);
20             }
21         }
22     }
23
24     static void add(MyList list, int count) {
25         for (int i = 0; i < count; ++i)
26             list.add(new MyInt(i));
27     }
28
29     static void print(MyList list) {
30         System.out.printf("Count: %d\n", list.getCount());
31         for (int i = 0; i < list.getCount(); ++i)
32             System.out.printf("%s ", list.getAt(i));
33         System.out.println();
34     }
35
36     static void doSomething(MyList list) throws Exception {
37         List<Thread> threads = new ArrayList<>();
38         add(list, 100);
39         for (int i = 0; i < 100; ++i) {
40             Thread thread = new Worker(new MyUnmodifiableList(list));
41             thread.start();
42             threads.add(thread);
43         }
44         for (Thread thread : threads)
45             thread.join();
46         print(list);
47     }
48
49     public static void main(String[] args) throws Exception {
50         doSomething(new MyArrayList());
51         doSomething(new MyLinkedList());
52     }
53 }
```