

## Supplementary Exercise on Memento Pattern

- (a) Draw a class diagram to show Memento Design Pattern.
- (b) State and briefly explain the roles in the Memento Design Pattern with the classes in your diagram.
- (c) You are asked to apply the Memento Design Pattern to save and restore the state of objects of the *Watch* class defined below. Implement a Java class *WatchMemento* with the constructor and *restore* methods which are used to respectively save and restore the state of a *Watch* object.

```
public class Watch{  
    String modelNo;      //watch model number  
    String name;        //watch name  
    int price;          //watch's unit price  
  
    public Watch(String modelNo, String name, int price){  
        setModelNo(modelNo);  
        setName(name);  
        setPrice(price)  
    }  
  
    public void setModelNo(String modelNo){  
        this.modelNo = modelNo;  
    }  
  
    public void setName(String name){  
        this.name = name;  
    }  
  
    public void setPrice(int price){  
        this.price = price;  
    }  
  
    public String toString() {  
        return name + ", " + price;  
    }  
}  
// class Watch
```

- (d) Write a *CareTaker* class to use the *WatchMemento* class implemented in your answer to Q2 (c). Your *CareTaker* class must provide three methods: constructor, *saveWatch* and *undo*. The *saveWatch* method saves the state of a *Watch* object. The *undo* method restores the associated *Watch* object to its previous state.

(e) Write a test program named *TestMemento.java* to test the *CareTaker* class and the *WatchMemento* class implemented in parts (c) and (d). Apply the *CareTaker* class in the test program so that all the update operations are undoable. Your test program should do the following steps in sequence:

- create a *CareTaker* object
- create a *Watch* object with *modelNo* = “A123”, *name* = “Solar Watch” and *price* = 3000
- save the state of the *Watch* object by calling the *saveWatch* method of the *CareTaker* object
- change the name of the *Watch* object to “Next Solar Watch”
- save the state of *Watch* object by calling the *saveWatch* method of the *CareTaker* object
- change the price of *Watch* object to 3500
- restore the previous state of the *Watch* object by calling the *undo* method of the *CareTaker* object
- print the name and the price of the *Watch* object on DOS command prompt screen