Memento Pattern

The **Memento Pattern** is used to restore a past state of an object. It is a **behavioural pattern**.

public class Memento {  
 private String state;  
  
 public Memento(String state) {  
 this.state = state;  
 }  
  
 public String getState() {  
 return state;  
 }  
}  
  
public class Originator {  
 private String state;  
  
 public void setState(String state) {  
 this.state = state;  
 }  
  
 public String getState() {  
 return state;  
 }  
  
 public Memento saveStateToMemento() {  
 return new Memento(state);  
 }  
  
 public void getStateFromMemento(Memento memento) {  
 state = memento.getState();  
 }  
}

public class CareTaker {  
 private *List*<Memento> mementoList = new ArrayList<Memento>();  
  
 public void add(Memento state) {  
 mementoList.add(state);  
 }  
  
 public Memento get(int index) {  
 return mementoList.get(index);  
 }  
}  
  
public class Demo {  
 public static void main(String[] args) {  
  
 Originator originator = new Originator();  
 CareTaker careTaker = new CareTaker();  
  
 originator.setState("State #1");  
 originator.setState("State #2");  
 careTaker.add(originator.saveStateToMemento());  
  
 originator.setState("State #3");  
 careTaker.add(originator.saveStateToMemento());  
  
 originator.setState("State #4");  
 System.*out*.println("Current State: " + originator.getState());  
  
 originator.getStateFromMemento(careTaker.get(0));  
 System.*out*.println("First saved State: " + originator.getState());  
 originator.getStateFromMemento(careTaker.get(1));  
 System.*out*.println("Second saved State: " + originator.getState());  
 }  
}

JAVA

