

بررسی الگوهای طراحی - Memento

توسط : محمد حسین زارع

@zare88r : twitter
telegram

Join Us : @JavaLandCH

مفاهیم

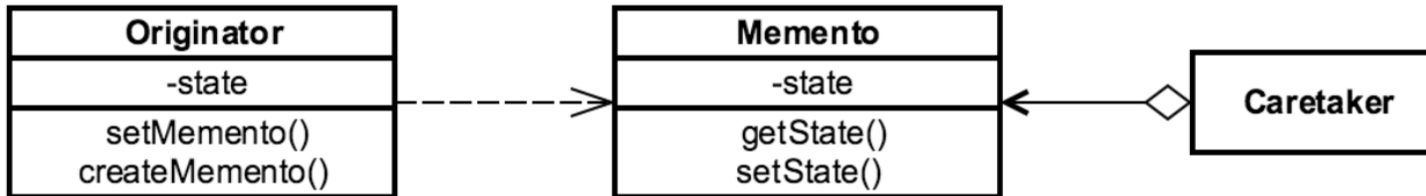
- برگرداندن شیء به حالت قبلی
- **Undo / Rollback**
- خروجی گرفتن از وضعیت داخلی اشیاء
- نمونه ها :
- **java.util.Date**
- **java.io.Serializable**

طراحی

Originator ▪

Caretaker ▪

Memento ▪



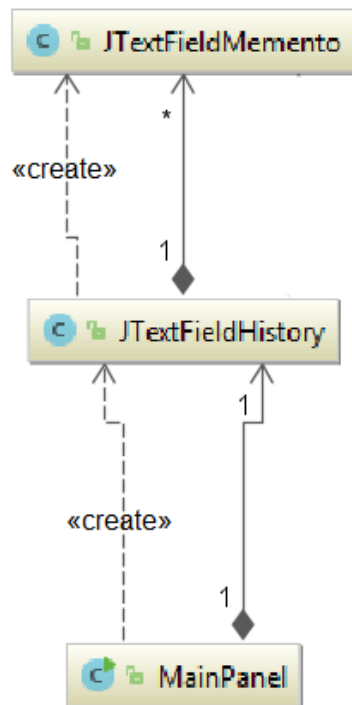
مثال 1

```
static class Data implements Serializable {  
    String value, extra;  
    int id;  
  
    private File storedFile;  
  
    public void savePoint() {  
        try {  
            storedFile = File.createTempFile("data", "obj");  
            try (ObjectOutputStream stream = new ObjectOutputStream(  
                new FileOutputStream(storedFile))) {  
                stream.writeObject(this);  
            }  
        } catch (IOException e) {  
            throw new IllegalStateException(e);  
        }  
    }  
}
```

مثال 1

```
public Data restoreSavePoint() {  
    if (storedFile != null) {  
        try (ObjectInputStream stream =  
            new ObjectInputStream(  
                new FileInputStream(storedFile));) {  
            return (Data) stream.readObject();  
        } catch (Exception e) {  
            throw new IllegalStateException(e);  
        }  
    } else {  
        throw new IllegalStateException("No Record has Saved yet");  
    }  
}
```

یک نمونه



یک نمونه

```
public class JTextFieldMemento {  
    private String text;  
  
    public JTextFieldMemento(String text) {  
        this.text = text;  
    }  
  
    public String getText() {  
        return text;  
    }  
  
    public void setText(String text) {  
        this.text = text;  
    }  
}
```

یک نمونه

//Caretaker

```
public class JTextFieldHistory {  
  
    private LinkedList<JTextFieldMemento> fieldMementos = new LinkedList<>();  
    private JTextField jTextField;  
  
    public JTextFieldHistory(JTextField jTextField) {  
        Objects.requireNonNull(jTextField);  
        this.jTextField = jTextField;  
    }  
  
    public void save() {  
        fieldMementos.push(new JTextFieldMemento(jTextField.getText()));  
        jTextField.setText("");  
    }  
  
    public void restore() {  
        if (!fieldMementos.isEmpty()) {  
            JTextFieldMemento pop = fieldMementos.pop();  
            jTextField.setText(pop.getText());  
        } else {  
            jTextField.setText("");  
        }  
    }  
}
```


یک نمونه

```
public MainPanel() {
    jLabel = new JLabel("Enter Your Name (Ctrl-Z to undo):");
    jTextField = new JTextField(20);
    jButton = new JButton("Save!");
    jTextFieldHistory = new JTextFieldsHistory(jTextField);

    setLayout(new FlowLayout());
    add(jLabel);
    add(jTextField);
    add(jButton);
    jButton.addActionListener(e -> jTextFieldHistory.save());

    jTextField.addKeyListener(new KeyAdapter() {
        @Override
        public void keyPressed(KeyEvent e) {
            if ((e.getKeyCode() == KeyEvent.VK_Z)
                && ((e.getModifiers() & KeyEvent.CTRL_MASK) != 0)) {
                jTextFieldHistory.restore();
            }
        }
    });
}
```

جمع بندی

- الگویی برای نگه داری تاریخچه
- ممکن است هزینه بر باشد
- مدیریت تاریخچه با برنامه نویس میباشد