

1. Model

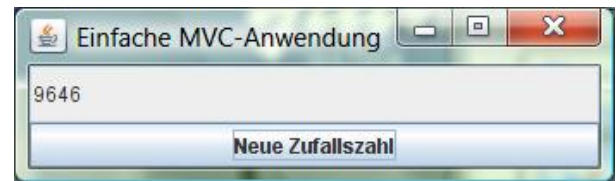
```
public class Model extends Observable
    implements IModel {
    private int zahl;

    public int getZahl() {return zahl;}

    public void setZahl(int zahl) {
        this.zahl = zahl;
        setChanged();
        notifyObservers(this);
    }

    @Override
    public void neueZahl() { this.setZahl((int) (Math.random() * 10000)); }

    @Override
    public void addObserver(Object o) { addObserver(o); }
}
```



2. View

```
public class View extends JFrame implements Observer {

    // Komponenten
    private Controller myController;
    private JTextField tfZufall = new JTextField();
    private JButton btnNeu = new JButton("Neue Zufallszahl");

    // Konstruktor
    public View(Controller controller) {
        super("Einfache MVC-Anwendung");
        this.controller = controller;
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        initComponents();
        setVisible(true);
    }

    // Komponenten initialisieren
    private void initComponents() {
        tfZufall.setEditable(false);
        tfZufall.setSize(80, 24);
        btnNeu.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                myController.neueZahl();
            }
        });
        add(btnNeu, BorderLayout.SOUTH);
        add(tfZufall, BorderLayout.CENTER);
        pack();
    }

    // Update - Methode (aus Observer-Interface)
    public void update(Observable o, Object arg) {
        Model m = (Model) arg;
        tfZufall.setText(String.valueOf(m.getZahl()));
    }
}
```

3. Interfaces

```
public interface IController { public void neueZahl(); }
```

```
public interface IModel {  
    public void addObserver(Object o);  
    public void neueZahl();  
}
```

4. Controller

```
public class Controller implements IController{  
    private IModel myModel;  
    private View myView;  
  
    public Controller() {  
        myView = new View(this);  
        myModel = new Model();  
        myModel.addObserver(myView);  
    }  
  
    @Override  
    public void neueZahl() {  
        myModel.neueZahl();  
    }  
}
```

5. Main

```
public class Main{  
    public static void main(String[] args) {  
        new Controller();  
    }  
}
```

6. Klassendiagramm:

