

GUI

MVC

P-Aufgaben



Folien: go.tum.de/904005

GUI

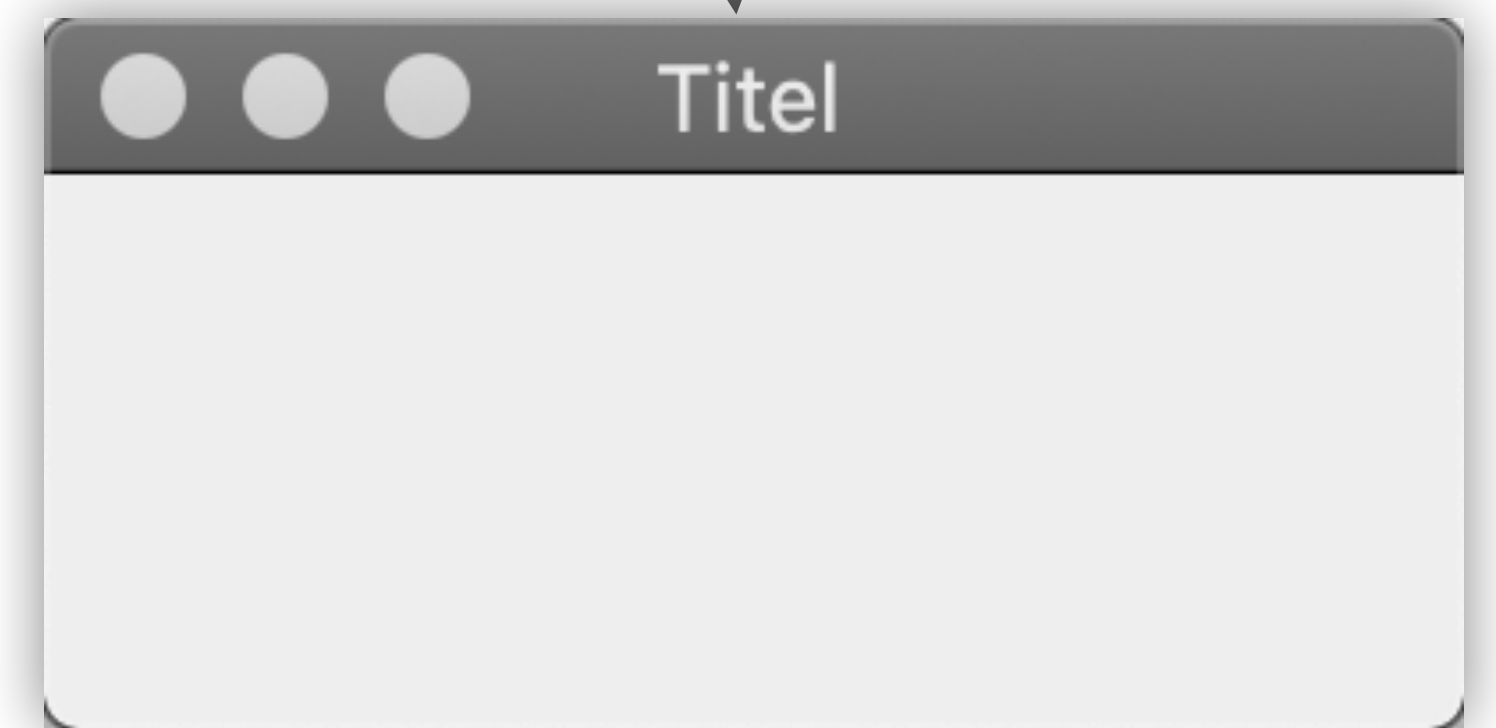
GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: JFrame

```
JFrame fenster = new JFrame();  
fenster.setTitle("Titel");  
fenster.setDefaultCloseOperation(  
    JFrame.EXIT_ON_CLOSE);  
fenster.setSize(200, 100);  
  
fenster.setVisible(true);
```



GUI

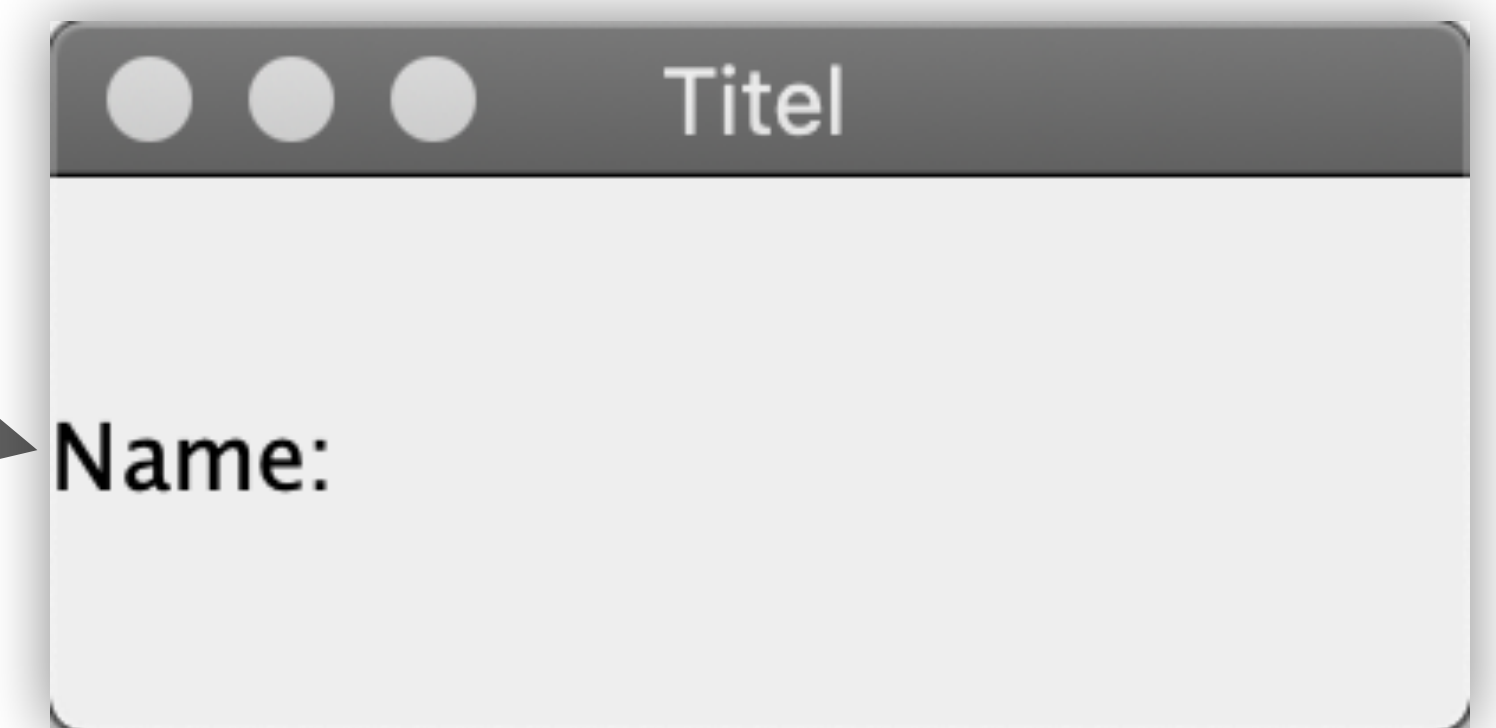
GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: JTextField

```
JLabel text = new JLabel();  
text.setText("Name:");  
fenster.add(text);
```



GUI

GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: JTextArea

```
JTextField eingabe = new JTextField();  
eingabe.setText("<name>");  
fenster.add(eingabe);
```



GUI

Graphical User Interface, Swing: JButton

```
JButton knopf = new JButton();  
knopf.setText("klick");  
fenster.add(knopf);
```



GUI

GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: LayoutManager

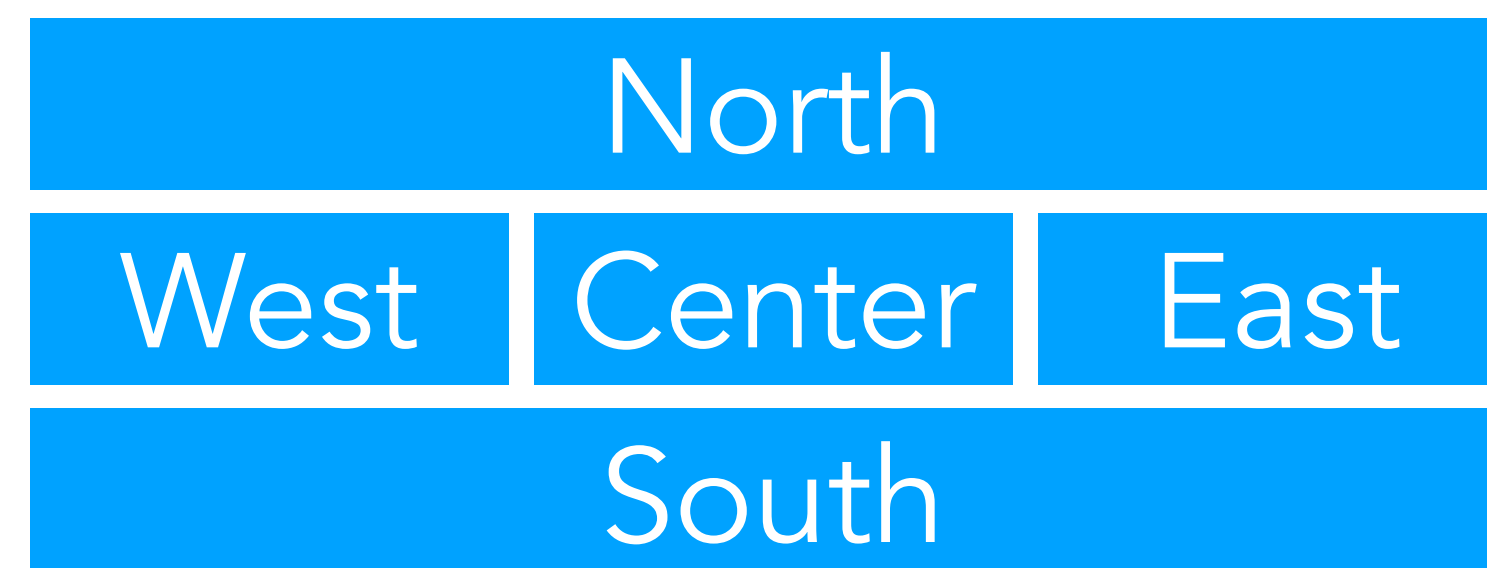
- Nicht Responsive (Veränderung der Fenstergröße)
- Alle Elemente liegen übereinander

```
fenster.setLayout(new BorderLayout());
```

```
fenster.add(text, BorderLayout.WEST);
```

```
fenster.add(eingabe, BorderLayout.EAST);
```

```
fenster.add(knopf, BorderLayout.SOUTH);
```



Übersicht:

<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

GUI

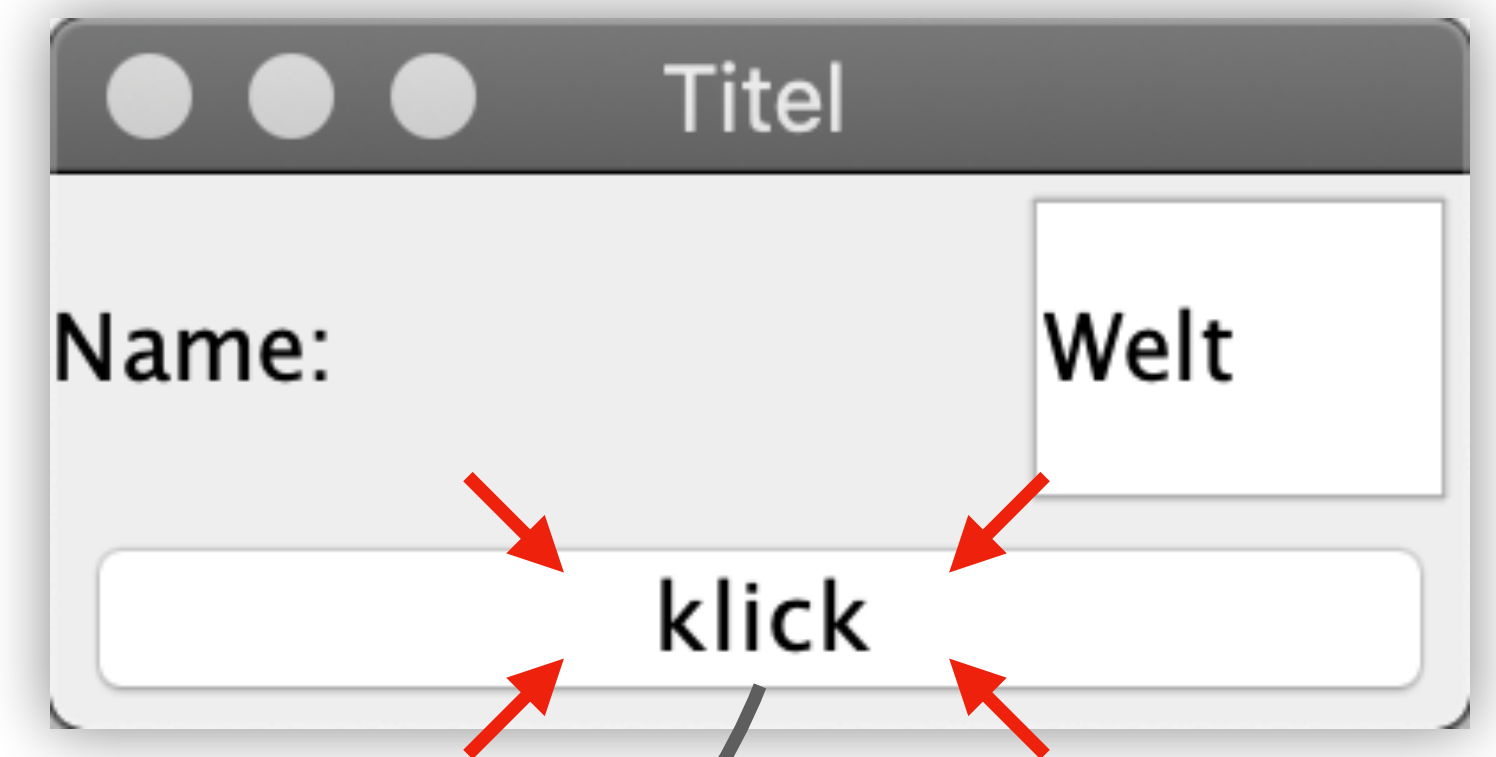
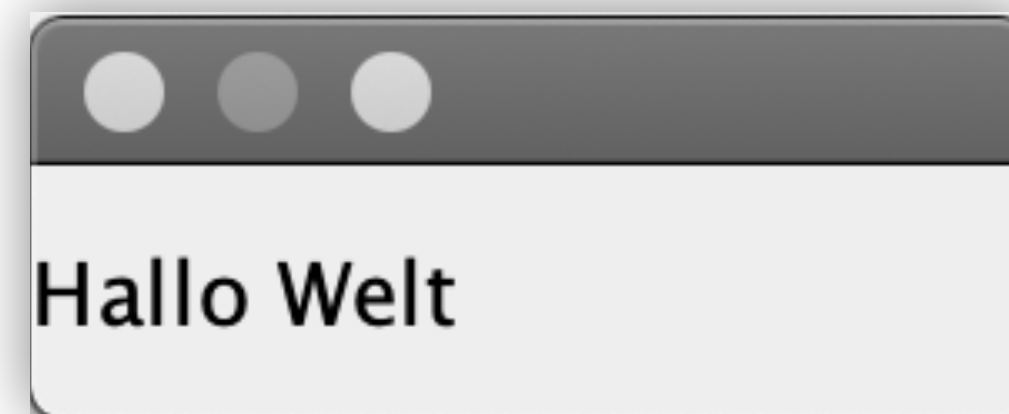
GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: JButton & ActionListener

```
knopf.addActionListener((event) -> {  
    JDialog dialog = new JDialog();  
    dialog.add(new JLabel("Hallo " +  
        eingabe.getText()));  
    dialog.setSize(150, 60);  
    dialog.setVisible(true);  
});
```



GUI

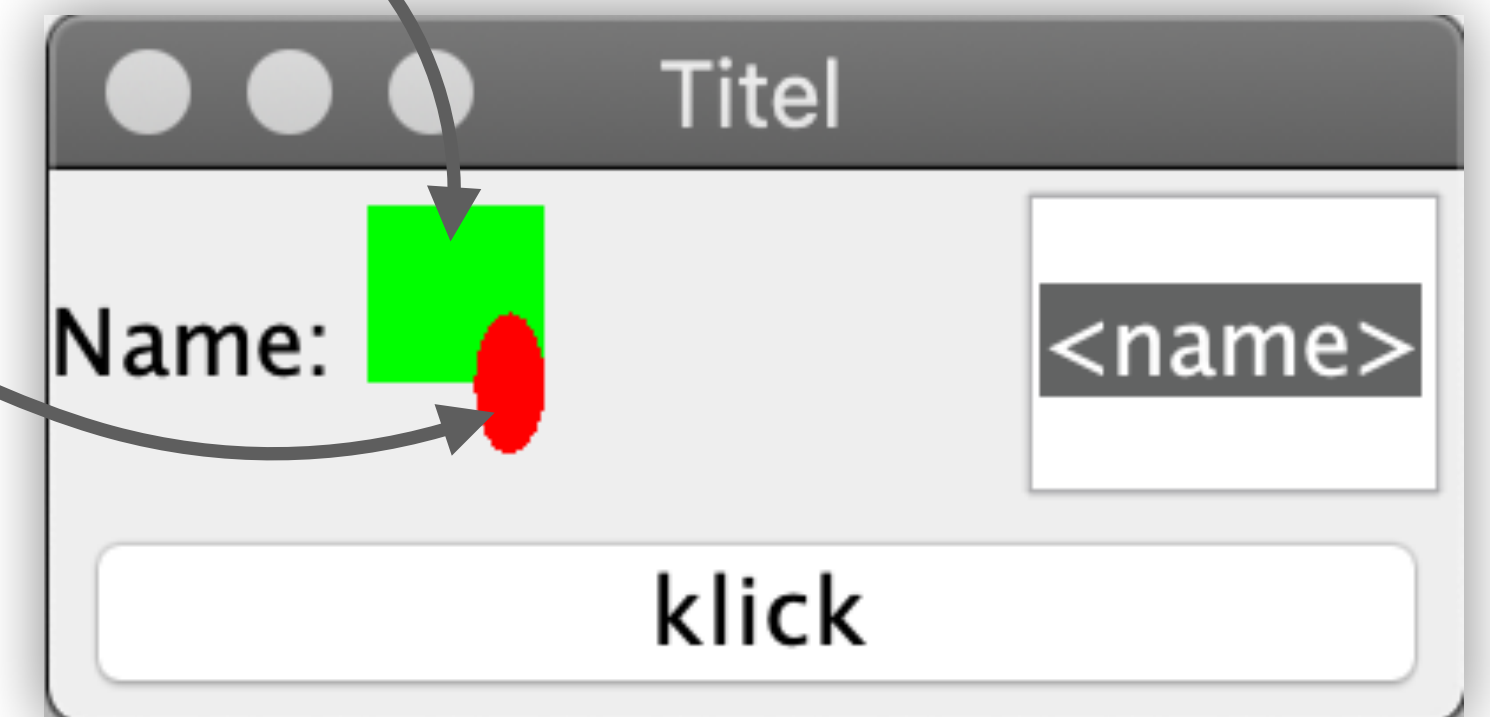
GUI

MVC

P-Aufgaben

Graphical User Interface, Swing: JPanel & Paint

```
JPanel leinwand = new JPanel() {  
    protected void paintComponent(Graphics g) {  
        g.setColor(Color.GREEN);  
        g.fillRect(5, 5, 25, 25);  
        g.setColor(Color.RED);  
        g.fillOval(20, 20,  
            10, 20);  
    }  
};
```



```
fenster.add(leinwand, BorderLayout.CENTER);
```


MVC

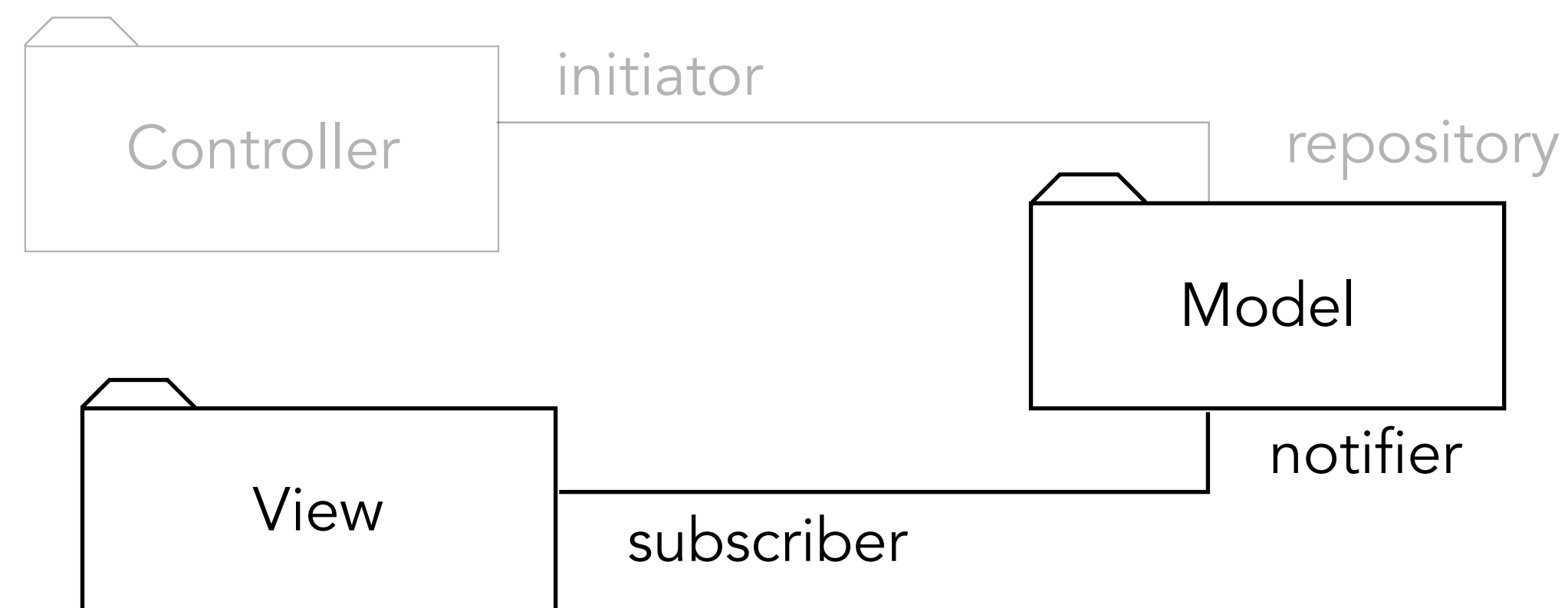
GUI

MVC

P-Aufgaben

The MVC style decouples data access (entity) and data presentation (boundary) nonhierarchically.

- View: GUI, display information to user
- Model: logics of the system

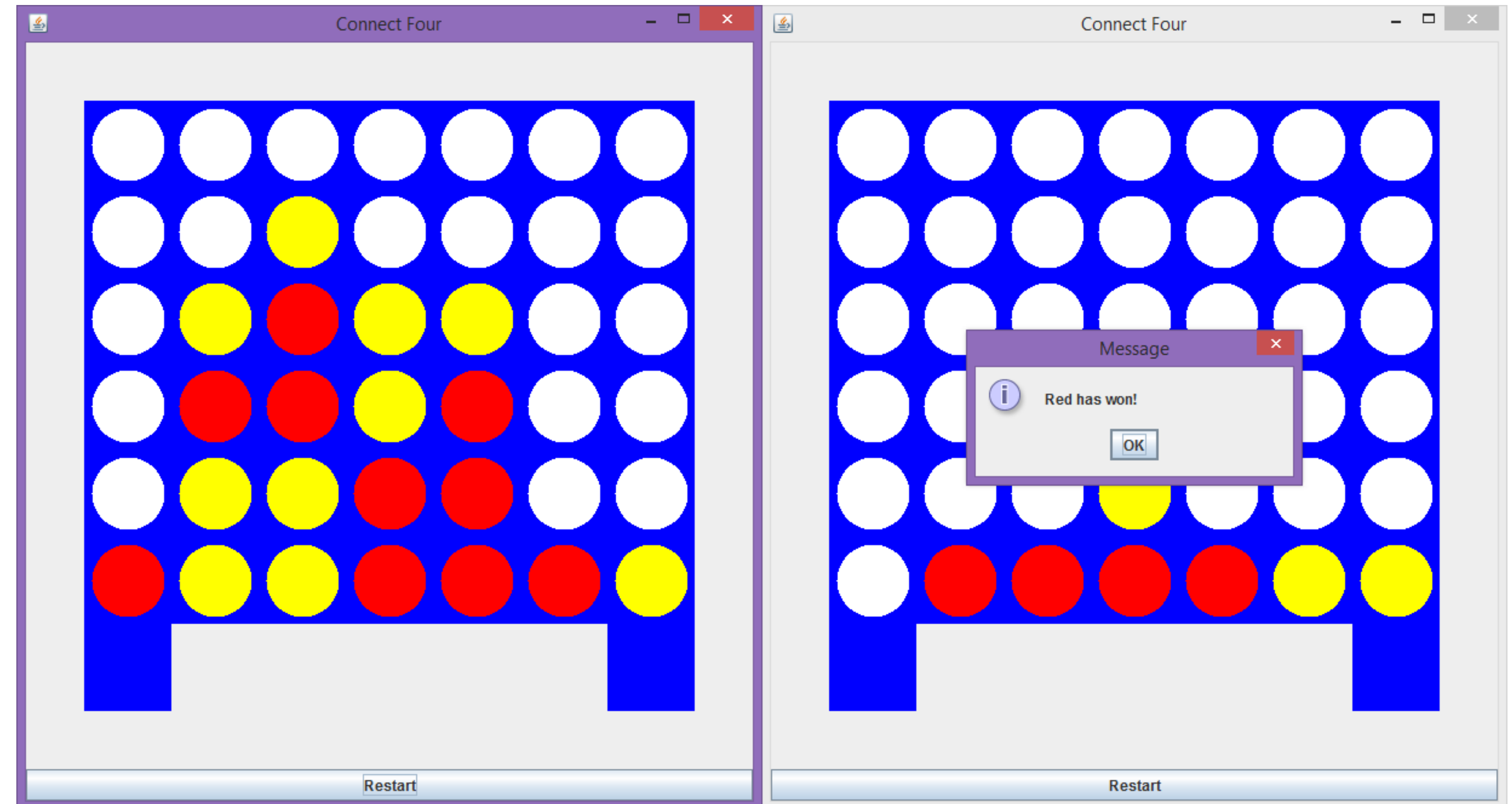


GUI

MVC

P-Aufgaben

4-Gewinnt



P14.01

GUI

MVC

P-Aufgaben

4-Gewinnt

- View, Fenster erstellen
- Model, Game Logik
- View und Model verknüpfen