

Bereich: Grafische Benutzeroberflächen (GUI/Swing) (1)**Swing-Grundkomponenten****Musterlösung****Package:** de.dhbwka.java.exercise.ui**Klasse:** ComponentFrame

```
package de.dhbwka.java.exercise.ui;

import java.awt.FlowLayout;
import javax.swing.*;

/**
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016-2018 by W. Geiger, T. Schlachter, C. Schmitt, W. Suess
 *
 * @author DHBW lecturer
 * @version 1.1
 */
@SuppressWarnings( "serial" )
public class ComponentFrame extends JFrame {

    public ComponentFrame() {
        this( "ComponentFrame" );
    }

    @SuppressWarnings( { "unchecked", "rawtypes" } )
    public ComponentFrame( String title ) {
        super( title );
        // Default layout for JFrame content would be BorderLayout!
        this.setLayout( new FlowLayout() );
        this.add( new JLabel( "JLabel" ) );
        this.add( new JTextField( "JTextField" ) );
        this.add( new JPasswordField( "JPasswordField" ) );
        this.add( new JButton( "JButton" ) );
        this.add( new JToggleButton( "JToggleButton" ) );
        this.add( new JCheckBox( "JCheckBox" ) );
        // No generic type for JComboBox - we had no generics yet
        this.add( new JComboBox(
            new String[] { "Item 1", "Item 2", "Item 3", "Item 4" } ) );
        // ButtonGroup connects the RadioButtons as a functional unit
        ButtonGroup group = new ButtonGroup();
        for ( int i = 1; i <= 3; i++ ) {
            JRadioButton rb = new JRadioButton( "Radio-Button-" + i );
            group.add( rb );
            this.add( rb );
        }
        this.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
        this.setSize( 640, 100 );
        this.setVisible( true );
    }

    public static void main( String[] args ) {
        new ComponentFrame( "Several basic Swing components" );
    }
}
```

Bereich: Grafische Benutzeroberflächen (GUI/Swing) (1)**Frame mit Text****Musterlösung****Package:** de.dhbwka.java.exercise.ui**Klasse:** TextFrame

```
package de.dhbwka.java.exercise.ui;

import java.awt.BorderLayout;
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

import javax.swing.JFrame;
import javax.swing.JTextArea;

/**
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016-2018 by W. Geiger, T. Schlachter, C. Schmitt, W. Suess
 *
 * @author DHBW lecturer
 * @version 1.1
 */
@SuppressWarnings( "serial" )
public class TextFrame extends JFrame {

    public TextFrame( String filename, int width, int height ) {
        super( filename );
        this.setLayout( new BorderLayout() );
        this.add( new JTextArea( TextFrame.readFile( filename ) ) );
        this.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
        this.setSize( width, height );
        this.setVisible( true );
    }

    /**
     * Read a text file and return its content as String
     *
     * @param filename
     *         file name to open text file from
     * @return content of file
     */
    private static String readFile( String filename ) {
        StringBuilder content = new StringBuilder();
        try ( BufferedReader br =
            new BufferedReader( new FileReader( filename ) ) ) {
            while ( br.ready() ) {
                content.append( br.readLine() ).append( System.lineSeparator() );
            }
        } catch ( IOException e ) {
            return "Error reading File " + filename;
        }
        return content.toString();
    }
}
```

```
public static void main( String[] args ) {  
    try { // if argument count < 3 or non-parsable content => exception  
        new TextFrame( args[0], Integer.parseInt( args[1] ),  
                        Integer.parseInt( args[2] ) );  
    } catch ( Exception e ) {  
        System.out  
            .println( "Please use: TextFrame <filename> <width> <height>\n" +  
                    "with an existing (relative or absolute) filename and\n" +  
                    "integer values for width and height." );  
    }  
}
```