

**Hinweis:** Es sind alle Felder auszufüllen! Abgabe der Übungsblätter immer **mittwochs** (Ausnahme wenn Feiertag: donnerstags) bis **spätestens 12:00 Uhr** in die entsprechend gekennzeichneten Briefkästen der Veranstaltung im Erdgeschoss des Instituts für Informatik (Gebäude N). Zuwiderhandlung wird mit Strafe geahndet! (Punktabzug)

Übungsblatt	
-------------	--

(hier die Nummer des bearbeiteten **Übungsblatts** eintragen)

	Übung 01 (1057 N) Montag 08:15 - 09:45 Uhr (Isabell Rücker)
	Übung 02 (1056 N) Montag 14:00 - 15:30 Uhr (Henning Cui)
	Übung 03 (1057 N) Montag 15:45 - 17:15 Uhr (Josef Kircher)
	Übung 04 (1054 N) Montag 17:30 - 19:00 Uhr (Mosaab Slimani)
	Übung 05 (1057 N) Montag 17:30 - 19:00 Uhr (David Hacker)
	Übung 06 (1055 N) Dienstag 12:15 - 13:45 Uhr (André Schweiger)
X	Übung 07 (1054 N) Dienstag 17:30 - 19:00 Uhr (Benjamin Sertolli)
	Übung 08 (1057 N) Dienstag 17:30 - 19:00 Uhr (Dat Le Thanh)
	Übung 09 (1054 N) Mittwoch 08:15 - 09:45 Uhr (Erik Pallas)
	Übung 10 (1055 N) Mittwoch 08:15 - 09:45 Uhr (Moritz Feldmann)
	Übung 11 (1054 N) Mittwoch 10:00 - 11:30 Uhr (Denise Böhm)
	Übung 12 (1056 N) Donnerstag 08:15 - 09:45 Uhr (Florian Magg)
	Übung 13 (1054 N) Donnerstag 15:45 - 17:15 Uhr (Marvin Drexelius)
	Übung 14 (1054 N) Donnerstag 17:30 - 19:00 Uhr (Patrick Eckert)
	Übung 15 (1057 N) Donnerstag 17:30 - 19:00 Uhr (Alexander Szöke)
	Übung 16 (1057 N) Freitag 08:15 - 09:45 Uhr (Philipp Braml)
	Übung 17 (1054 N) Freitag 10:00 - 11:30 Uhr (Elisabeth Korndörfer)
	Übung 18 (1054 N) Freitag 12:15 - 13:45 Uhr (Philipp Häusele)
	Übung 19 (1056 N) Freitag 12:15 - 13:45 Uhr (Maximilian Demmler)
	Übung 20 (1054 N) Freitag 14:00 - 15:30 Uhr (Florian Straßer)

(hier die eingeteilte **Übungsgruppe** ankreuzen)

Teamnummer	6
------------	---

(hier die Nummer des eingeteilten **Teams** eintragen)

Tarik Selimovic
Anton Lydike
Dominic Cesnak

(hier die **Vor- und Nachnamen** aller Teammitglieder eintragen)

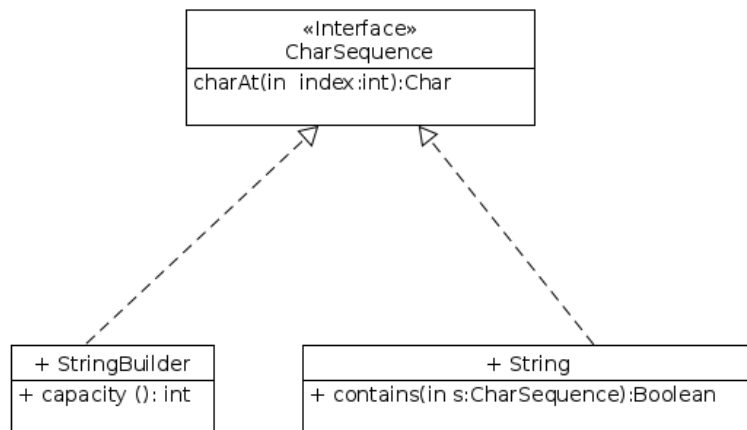
Aufgabe		
Aufgabe		
Aufgabe		
Aufgabe		
<b>Gesamt</b>		

(vom Tutor auszufüllen)

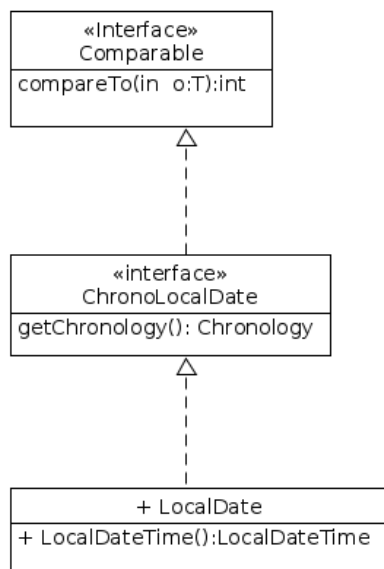
# Übungsblatt 6

21)

a)



b)



**22)****a)**

```
package aufgabe22;

import java.awt.*;
import java.awt.event.*;

public class A extends Frame {
    private static final long serialVersionUID = 1L;

    public A () {
        super("Aufgabe22a");
        GridLayout grid = new GridLayout(3,2);
        grid.setHgap(4);

        Label labels[] = {
            new Label("Vorname:", Label.RIGHT),
            new Label("Name:", Label.RIGHT),
            new Label("Ort:", Label.RIGHT)
        };

        for (Label label : labels) {
            label.setForeground(Color.BLUE);
            label.setFont(new Font("Arial", Font.BOLD, 16));
            this.add(label);
            this.add(new TextField());
        }

        this.setLayout(grid);
        this.pack();
        this.setVisible(true);

        this.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent we) {
                dispose();
            }
        });
    }

    public static void main(String[] args) {
        new A();
    }
}
```

**b)**

```
package aufgabe22;

import java.awt.*;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

public class B extends Frame {
    private static final long serialVersionUID = 1L;

    public B () {
        super("Aufgabe22b");
        FlowLayout flow = new FlowLayout();

        Label label = new Label("Anzahl auswählen:");
        label.setFont(new Font("Arial", Font.BOLD, 16));

        Choice choice = new Choice();
        for (int i = 0; i <= 9; i++) {
            choice.add(String.valueOf(i));
        }
        choice.select(1);

        this.add(label);
        this.add(choice);

        this.setLayout(flow);
        this.pack();
        this.setVisible(true);

        this.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent we) {
                dispose();
            }
        });
    }
    public static void main(String[] args) {
        new B();
    }
}
```

**23)****a)**

```
package aufgabe23;

import java.awt.*;
import java.awt.event.*;

public class A extends Frame implements KeyListener {
    private static final long serialVersionUID = 1L;

    public A () {
        super("aufgabe23a");
        this.setVisible(true);
        this.setSize(new Dimension(200, 200));
        this.addKeyListener(this);
    }

    public void keyReleased(KeyEvent e) {
        if (e.getKeyChar() == 'x') {
            this.dispose();
        }
    }

    @Override
    public void keyTyped(KeyEvent e) {}

    @Override
    public void keyPressed(KeyEvent e) {}

    public static void main(String[] args) {
        new A();
    }
}
```

b)

```
package aufgabe23;

import java.awt.*;
import java.awt.event.*;

public class B extends Frame implements ItemListener {
    private static final long serialVersionUID = 1L;
    private TextField t;

    public B () {
        super("aufgabe23b");

        BorderLayout bl = new BorderLayout();
        FlowLayout fl = new FlowLayout();
        Choice c = new Choice();
        Panel p = new Panel();
        t = new TextField();

        this.setLayout(bl);
        this.add(p, BorderLayout.NORTH);
        this.add(t, BorderLayout.SOUTH);

        p.setLayout(fl);
        p.add(new Label("Bemerkung"));
        p.add(c);

        c.add("ja");
        c.add("nein");
        c.select(1);
        c.addItemListener(this);

        t.setVisible(false);

        this.pack();
        this.setVisible(true);

        this.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent we) {
                dispose();
            }
        });
    }

    public void itemStateChanged (ItemEvent e) {
        t.setVisible(e.getItem() == "ja");
        this.pack();
    }

    public static void main(String[] args) {
        new B();
    }
}
```

**24)****a)**

```
package aufgabe24;

import java.awt.*;
import java.awt.event.*;

public class A extends Frame {
    private static final long serialVersionUID = 1L;

    public A () {
        super("");
        this.setVisible(true);
        this.setSize(new Dimension(500, 50));

        A a = this;

        this.addWindowListener(new WindowAdapter() {
            int tries = 0;

            public void windowClosing(WindowEvent we) {
                a.setTitle(String.valueOf(++tries));
                if (tries == 3) {
                    dispose();
                }
            }
        });
    }

    public static void main(String[] args) {
        new A();
    }
}
```

**b)**

```
package aufgabe24;

import java.awt.*;
import java.awt.event.*;

public class B extends Frame implements ActionListener {
    private static final long serialVersionUID = 1L;
    double size = 100;
    Button larger = new Button("+");
    Button smaller = new Button("-");

    public B () {
        super("");

        GridLayout grid = new GridLayout(2,1);
        this.setLayout(grid);
        this.add(larger);
        this.add(smaller);

        larger.addActionListener(this);
        smaller.addActionListener(this);

        this.setTitle(String.valueOf(size));
        this.setSize(new Dimension((int) size, (int) size));
        this.setVisible(true);

        this.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent we) {
                dispose();
            }
        });
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource().equals(larger)) {
            size *= 1.1;
        } else if (e.getSource().equals(smaller)) {
            size *= 0.9;
        }

        this.setSize(new Dimension((int) size, (int) size));
        this.setTitle(String.valueOf((int) size));
    }

    public static void main(String[] args) {
        new B();
    }
}
```