

FACHHOCHSCHULE RAVENSBURG-WEINGARTEN
STUDIENGANG ANGEWANDTE INFORMATIK

Prüfung 1884,1812,4018(WI)

Grafische Bedienoberflächen / Teil 2

Fachgebiet:	Grafische Bedienoberflächen
Prüfer:	Prof. Dr.-Ing. S. Keller
Datum:	4.2.2003
Ort:	C004
Zeit	14:00 - 15.30
Dauer :	90 min (Teil 1 + Teil 2)
Aufgabenblätter:	8 (Teil 2)
erreichbare Punktzahl:	45 (Teil 2)
zugelassene Hilfsmittel:	Ausgeteiltes Material + handschriftliche Notizen

Hinweise zur Niederschrift der Lösungen:

- Schreiben Sie Name und Matrikelnummer auf jedes Aufgabenblatt !!
- Schreiben Sie **lesbar** !!!
Wenn der Prüfer Ihre Angaben nicht lesen kann verschenken Sie wertvolle Punkte !!!

Name:

Matrikelnummer:

Realisierung einer grafischen Oberfläche unter Java

Sie sollen eine Java-Standalone-Anwendung mit GUI unter Verwendung von swing-Komponenten implementieren. Das folgende Bild visualisiert die zu realisierende Oberfläche.

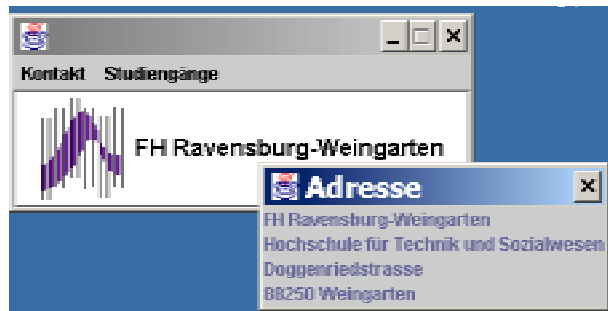


Beim Starten der Anwendung erscheint ein Fenster ohne Titel.

Im Fenster erscheint links das FH-Logo und rechts daneben der Schriftzug "FH Ravensburg-Weingarten". Der Text wird in dem Font Arial, Stil=fett, Grösse=16 und in einer von Java voreingestellten Farbe dargestellt. Der Hintergrund erscheint weiß. Das Fenster kann in seiner Größe nicht verändert werden und enthält eine Menüleiste mit den Menüpunkten „Kontakt“ und „Studiengänge“. Das FH-Logo und der Text ist eine auswählbare Komponente. (siehe Bild links).

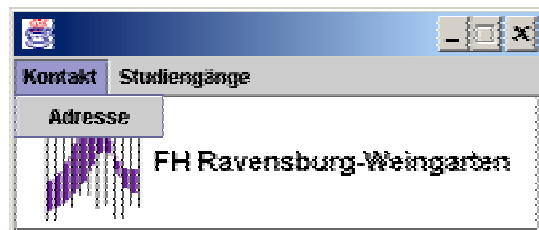
Bei Auswahl des FH-Logos erscheint ein Dialogfenster, in dem die Adresse der FH angezeigt wird. Das Dialogfenster ist modal und kann nur über den exit-Knopf (x) geschlossen werden.

(siehe Bild rechts)

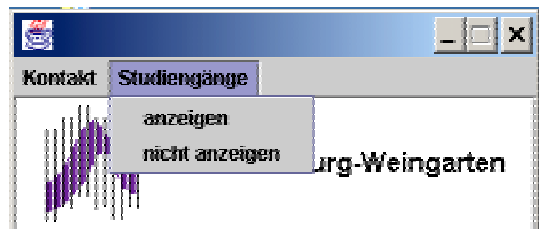


Bei Auswahl des Menüs „Kontakt“ erscheint ein Menüpunkt, „Adresse“ (siehe Bild rechts).

Bei Auswahl von „Adresse“ erscheint das gleiche Dialogfenster, wie bei Auswahl des FH-Logos. (siehe Bild oben)



Bei Auswahl des Menüs „Studiengänge“ erscheinen die Menüpunkte, „anzeigen“ und „nicht anzeigen“ (siehe Bild rechts).



Durch Auswahl des Menüpunktes „anzeigen“, erscheint unterhalb des Logos mit dem FH-Schriftzug ein Baum, der die drei Studiengängen AI, WI und IK der FH darstellt.

Durch Auswahl des Menüpunktes „nicht anzeigen“ verschwinden das angezeigte Bäumchen wieder.



Bei der Erstellung des Programms gehen Sie in zwei Phasen vor:

1. Phase: Erstellen der Anwendung ohne Menübar
2. Phase: Erweiterung der Anwendung um die Menübar mit den beiden Menüs „Kontakt und „Studiengänge“.

Aufgabe 1 (Phase 1)

(15 Punkte)

Das von ihnen erstellte JAVA-Programm besteht in der 1. Phase aus den 2 Javaklassen

fhstudium.java und logolistener.java.

Diese Objektklassen sind im Folgenden beschrieben. Dabei wurden alle import-Anweisungen aus Gründen der Übersichtlichkeit weggelassen. Gehen Sie davon aus, dass für alle Klassen alle benötigten import-Anweisungen vorhanden und korrekt sind.

Klasse: fhstudium.java

```
public class fhstudium extends JFrame

{
    JLabel logo;
    Container darstellungsflaeche;

    public fhstudium()
    {
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);

        this.setResizable(false);

        Icon bild=new ImageIcon(System.getProperty("user.dir")+"\\\\"+"fh.gif");

        logo=new JLabel("FH Ravensburg-Weingarten");

        logo.setBackground(Color.white);

        logo.setFont(new Font("Arial",Font.BOLD,16));

        darstellungsflaeche=this.getContentPane();

        darstellungsflaeche.setLayout(new BoxLayout(darstellungsflaeche,BoxLayout.Y_AXIS));

        darstellungsflaeche.setBackground(Color.white);

        darstellungsflaeche.add(logo);

        this.pack();

        this.show();

    }

    public static void main(String[] args)
    {
        fhstudium FH = new fhstudium();
    }
}
```

Klasse: logolistener.java

```
public class logolistener implements ActionListener

{
    JFrame vater;

    public logolistener(JFrame frame)

    {
        vater=frame;
    }
}
```

```
public void actionPerformed(ActionEvent e)

{
    JDialog adressfenster=new JDialog(vater,"Adresse");

    adressfenster.setDefaultCloseOperation(JDialog.DISPOSE_ON_CLOSE);

    adressfenster.setLocation(100,100);

    adressfenster.getContentPane().setLayout(new
BoxLayout(adressfenster.getContentPane(),BoxLayout.Y_AXIS));

    adressfenster.getContentPane().add(new JLabel("FH Ravensburg-Weingarten"));

    adressfenster.getContentPane().add(new JLabel("Hochschule für Technik und Sozialwesen"));

    adressfenster.getContentPane().add(new JLabel("Doggenriedstrasse"));

    adressfenster.getContentPane().add(new JLabel("88250 Weingarten"));

    adressfenster.pack();
}
}
```

Sie übersetzen und testen ihr Programm und stellen folgende Fehler fest:

- 1.1 Nach dem Starten des Programms erscheint ein Fenster, in dem nur der Text zu sehen ist. Das Logo fehlt. Gehen sie davon aus, dass die Bilddatei im korrekten Verzeichnis liegt.



Wieso fehlt das Logo? Erklären Sie die Fehlerursache.

Modifizieren Sie das oben angegebene Programm, so das Logo und Text wie in der Aufgabenstellung dargestellt erscheinen. (siehe Hinweise !!!!).

- 1.2 Gehen Sie davon aus, sie haben den Fehler von 1.1. korrigiert. Bei Anwahl des Logos wird aber kein Adressfenster angezeigt. Dies ist auf vier Fehler zurückzuführen. Diese Fehler sind in beiden Klassen versteckt. Erläutern Sie die gefundenen Fehler und ändern bzw. ergänzen sie das Programm, so das bei Anwahl des Logos das Dialogfenster auch tatsächlich erscheint (siehe Hinweise !!!!).
- 1.3 Das Dialogfenster enthält 4 Textzeilen, die in einem Boxlayout dargestellt werden. Welches Layout vom „awt“ hätte man alternativ zum Boxlayout (swing) nehmen können ?
- Wie muss man das Programm ändern ?

Aufgabe 2 (Phase 2)**(30 Punkte)**

In der 2. Phase erweitern Sie ihr Programm um die Menübar mit den beiden Menüs „Kontakt“ und „Studiengänge“. Sie implementieren dazu eine eigene Klasse FHMenu, eine Klasse baumlistener und eine Klasse studiengaenge. Die Klasse FHMenu realisiert die Menübar. Die Klasse baumlistener reagiert auf Auswahl der Menüpunkte Studiengänge anzeigen / nicht anzeigen. Die Klasse studiengaenge repräsentiert den kleinen Studiengangsbaum.

Klasse: FHMenu.java

```
public class FHMenu extends JMenuBar
{
    public FHMenu(
    { JMenu studiengaenge=new JMenu("Studiengänge");
      JMenuItem einschalten=new JMenuItem("anzeigen");
      JMenuItem abschalten=new JMenuItem("nicht anzeigen");

      JMenu Kontakt=new JMenu("Kontakt");
      JMenuItem Adresse=new JMenuItem("Adresse");

    }
}
```

Name:

Matrikelnummer:

Klasse: baumlistener.java

```
public class baumlistener implements ActionListener
{
    studiengaenge anzeige;

    public baumlistener()
    {
        anzeige=new studiengaenge();
    }

    public void actionPerformed(ActionEvent e)
    {
        JMenuItem source = (JMenuItem)e.getSource();

        String command=e.getActionCommand();

        fhstudium anwendung=(fhstudium)
(((JPopupMenu)source.getParent()).getInvoker().getParent().getParent().getParent());

        if ( command.equals("anzeigen")) anwendung.getContentPane().add(anzeige);
        else                             anwendung.getContentPane().remove(anzeige);

        anwendung.pack();
        anwendung.repaint();
        anwendung.validate();
    }
}
```

Klasse: studiengaenge.java

```
public class studiengaenge extends JPanel
{
    DefaultMutableTreeNode wurzel;
    DefaultTreeModel baum;

    public studiengaenge()
    {
        this.setBackground(Color.white);

    }
}
```

- 2.1 In der Klasse FHMenu fehlen noch einige Anweisungen, damit das Menü korrekt erscheint und die Menüpunkte auch tatsächlich auf eine Auswahl reagieren. Bitte beachten Sie dabei, dass bei Auswahl des Menüpunktes „Adresse“, nicht eine neue Listenerobjekt-Instanz erzeugt wird, sondern die schon vorhandene Listenerobjekt-Instanz verwendet wird, die auch bei Auswahl des FH-Logos aktiviert wird.

Ergänzen Sie die Klasse FHMenu mit den fehlenden Anweisungen.

Wie fügen Sie das Menü in ihre Anwendung ein ?

Dokumentieren Sie im Programm an welchen Stellen welche Ergänzungen vorzunehmen sind.

- 2.2 In der Klasse baumlistener wird das Objekt *anwendung* über die Methode getParent() ermittelt und nicht per Parameterübergabe an die Klasse übergeben. Prüfen Sie, ob „die Objektreferenz „anwendung“ korrekt berechnet wird oder ob ein Fehler auftritt. Begründen Sie Ihre Entscheidung, indem Sie für jede Stufe genau angeben, auf welches Objekt der Aufruf verweist. Eine Antwort ohne Begründung ergibt keine Punkte.

```
fhstudium anwendung=(fhstudium)
```

```
((JPopupMenu)source.getParent()).getInvoker().getParent().getParent().getParent().getParent());
```

Bitte hier
Objekt eintragen,
das aus der obigen
Methode geliefert
wird

--	--	--	--	--	--

In obiger Java-Anweisung wird das berechnete Ergebnis durch den cast-Operator in einen anderen Objekttyp gewandelt. Wieso ist dieser cast-Operator notwendig ?

- 2.3 In der Klasse studiengaenge soll sowohl der Baum für die Studiengänge als auch die Komponente zur Darstellung des Baumes erzeugt werden. Vervollständigen Sie die Klasse mit allen notwendigen Anweisungen, so dass bei Auswahl des Menüpunktes „Studiengänge anzeigen“ der Baum auch tatsächlich dargestellt wird.

Hinweise:

- Bei Änderungen markieren Sie bitte die Zeilen im Programm, wo und welche Änderung durchzuführen sind. Bei Programmergänzungen geben Sie die Programmergänzungen auf einem gesonderten Blatt an und kennzeichnen die Stelle im Programm an der die Ergänzung einzufügen ist. Gegeben Sie zu jeder Markierung, Änderung oder Ergänzung im Listing die Aufgabennummer z.B. 1.3 an.
- Sie müssen keine exakte Syntax benutzen. Es muß jedoch der Aufbau der Java-Anwendung d.h. die Objektklassen, deren Eigenschaften und Methoden, sowie die Instanziierungen zu Objekten erkennbar sein.