

Übung 1

Wiederholung JAVA

Hinweise: Das JAVA Development Kit (JDK) zum Entwickeln von JAVA-Programmen und das JAVA Runtime Environment (JRE) zum Ausführen von JAVA-Programmen finden Sie für verschiedene Betriebssysteme unter

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Die beliebte ECLIPSE Entwicklungsumgebung für Java finden Sie unter

<http://www.eclipse.org/>

1. (Vorbereitung)
2. Verifizieren Sie Ihre Java-Installation, indem Sie das altbekannte "**Hallo, Welt!**", das hier nicht noch einmal abgedruckt wird, übersetzen und ausführen.

*Datei **hello.java**:*

```
public class hello {  
    // Hauptmethode, wird von der JVM aufgerufen,  
    // wenn das Programm mit "java hello" gestartet wird.  
    public static void main(String[] args) {  
        // Methode println() aus Unterklasse out der  
        // Klasse System aufrufen  
        System.out.println("Hallo, Welt!");  
    }  
}
```

```
javac hello.java  
java hello  
Hallo, Welt!
```

3. Schreiben Sie ein Java-Programm, das das Ergebnis der Rechnung

$$1 + \frac{1}{3}$$

als

- (a) Ascii-Zeichen
- (b) Ganzzahl
- (c) einfachgenaue Fließkommazahl
- (d) doppeltgenaue Fließkommazahl

berechnet und ausgibt. Die Berechnung selbst soll auch mit dem jeweils angegebenen Basisdatentyp durchgeführt werden, nicht nur die Ausgabe. Wiederholen Sie den Versuch mit der Rechnung $\frac{1}{9}$. Was fällt bei der Ausgabe der Ergebnisse auf?

Datei **u1a3.java**:

```
// Übung 1 Aufgabe 3
public class u1a3 {
    public static void main(String[] args) {
        char c    = (char)1 + (char)1 / (char)3; // char-Konstante
        int  i    = 1 + 1 / 3;                 // int-Konstante
        float f   = 1.0F + 1.0F / 3.0F;        // float-Konstante
        double d  = 1.0 + 1.0 / 3.0;           // double-Konstante

        // Ohne (byte) cast wird das ZEICHEN 1 ausgegeben, nicht der
        // WERT 1.
        // System.out.println("1+1/3 als byte   ergibt: " + (byte)c);
        System.out.println("1+1/3 als byte   ergibt: " + c);
        System.out.println("1+1/3 als int    ergibt: " + i);
        System.out.println("1+1/3 als float  ergibt: " + f);
        System.out.println("1+1/3 als double ergibt: " + d);
    }
}
```

Vergl. Java-Basistypen:

http://de.wikipedia.org/wiki/Java-Syntax#Primitive_Datentypen

Die Ausgabe bei der float-Berechnung von $1 + \frac{1}{3}$ ist bereits auffällig. Auch interessant wäre $1 + \frac{1}{9}$, bei dem man die durch die binäre Berechnung der Mantisse auftretenden Rundungsfehler in den Nachkommastellen beobachten kann.

4. Bilden Sie die mathematische Funktion

$$f(x, y) = \begin{cases} 1 & \text{falls } x > y \\ -1 & \text{falls } x < y \\ 0 & \text{falls } x = y \end{cases}$$

als Java-**Funktion** mit **int**-Datentypen sowohl für Rückgabewert als auch Übergabe-Parameter ab.

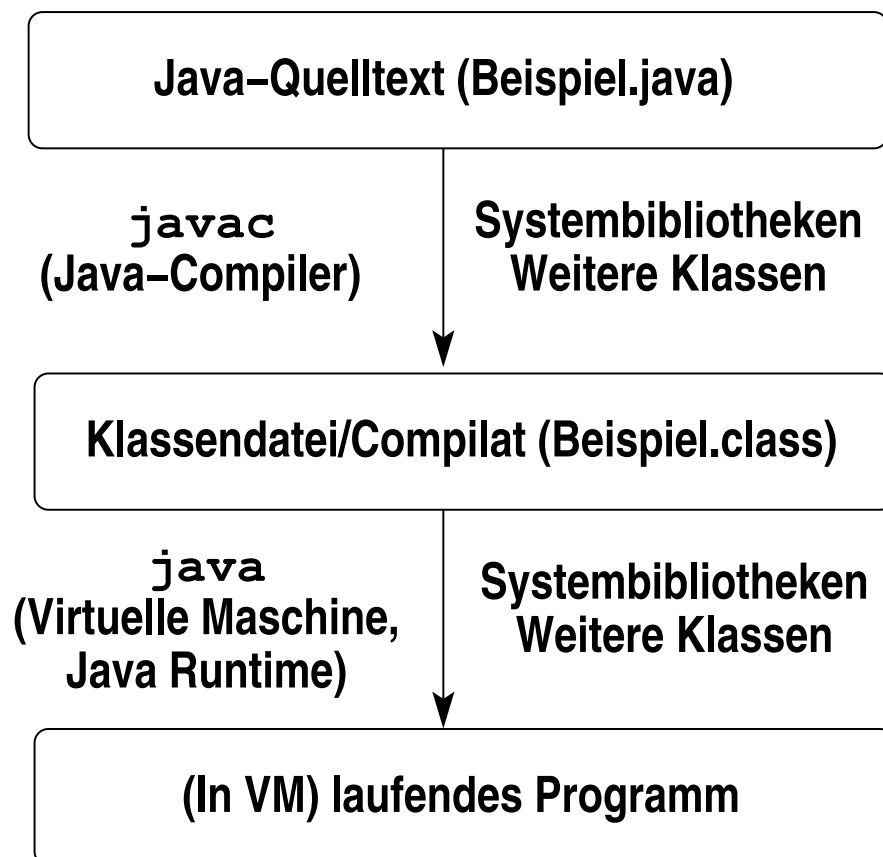
Datei **u1a4.java**:

```
public class u1a4 {
    public static int f(int x, int y) {
        if(x>y) return 1;
        else if(x<y) return -1;
        else return 0;
    }
    // Das "else" ist oben eigentlich in allen Fällen überflüssig,
    // da return die Methode bereits verlässt.

    // Alternativ: public int f(int x, int y) {
    //                 return (x>y)?1:(x<y)?-1:0; }
}
```

5. Erklären Sie anhand einer einfachen Skizze, wie man vom Java-Quelltext zu einem lauffähigen Programm kommt, und welche Java-Komponenten hierfür eingesetzt werden.

Vom Java-Quelltext zum lauffähigen Programm.



6. Mit Hilfe einer HTML-Seite wie dieser können Sie ein graphisches Java-Applet im Webbrowser ablaufen lassen, sofern das entsprechende Java-Plugin installiert und aktiviert ist (beachten Sie auch die Installations-Hinweise für das JAVA-Plugin für Ihr Betriebssystem und Ihren Web-Browser!).

Datei `applet-beispiel.html`:

```
<HTML>
<HEAD><TITLE>Beispiel für Grafik in Java-Applets</TITLE></HEAD>
<BODY>
```

Beispiel-Applet (Quelltext) von `LachendesGesicht.java`:

```
<PRE>

import java.awt.Graphics;
public class LachendesGesicht extends java.applet.Applet {
    public void paint(Graphics g) {
        g.drawOval(1, 1, 398, 398); // Zeichne einen Kreis
        g.fillOval(200-100, 180-40, 40, 20); // Auge links
        g.fillOval(200+ 60, 180-40, 40, 20); // Auge rechts
        g.drawLine(200,160,200,260); // Nase
        g.drawArc(80, 220, 240, 90, 360-90-60, 120); // Mund
    }
}

</PRE>
```

Hier ist der Bereich, in dem das Applet läuft:


```
<CENTER>
<APPLET CODE="LachendesGesicht.class" CODEBASE="."
        width=400 height=400>
</APPLET>
</CENTER>

</BODY>
</HTML>
```

Den Inhalt der Datei `LachendesGesicht.java` können Sie dem im HTML-Code angegebenen Java-Quelltext entnehmen, selbst wenn Sie (noch) kein HTML beherrschen.

Die Lösung ist in der Aufgabe integriert (cut & paste).

7. Welche Ausgabe liefert dieses JAVA-Fragment?

```
int result = 0;
```

```
for(int i=0; i<5; i++)
  switch(i){
    case 3:
      result += 10; break;
    default:
      result += i;
  }
System.out.println(result);
```

Im Zweifelsfall: Ausprobieren. :-)

8. Welche Ausgabe liefert dieses JAVA-Fragment?

```
int result = 0;
for(int i=0; i<5; i++)
  switch(i){
    case 3:
      result += 10;
    default:
      result += i;
  }
System.out.println(result);
```

*Falls Sie es nicht gesehen haben: Der Unterschied zur vorigen Aufgabe liegt im Kommando **break**;*

Im Zweifelsfall: Ausprobieren. :-)

9. Welche Ausgabe liefert dieses JAVA-Fragment?

```
String s1 = "I'm ";
String s2 = " years old.";
int age = 4; ++age; age += 14; age--;
System.out.println(s1 + age + s2);
```

Im Zweifelsfall: Ausprobieren. :-)