

# Einführung in die Programmierung

## Programmieren in Java

### Übung 1

#### Grundelemente eines Java-Programms

#### Aufgabe 1

Die Datei

uebung1.zip

enthält u. a. die Java-Quelldateien Pentium.java und Echo.java. Kopieren Sie diese Dateien in eines Ihrer Dateiverzeichnisse. Übersetzen Sie den Java-Quellcode mit dem Java-Compiler javac und lassen Sie das Programm im Java-Interpretierer java ablaufen.

#### Aufgabe 2

Geben Sie das folgende Programm im Editor ein und speichern Sie es in einer Datei namens FitzQuack.java ab (z. B. mit dem Kommando notepad FitzQuack.java). Übersetzen Sie das Programm mit dem Java-Compiler javac und lassen Sie es im Java-Interpretierer java ablaufen.

```
/**  
 * Das folgende Programm "spielt" das Spiel "FitzQuack".  
 * Das Programm zaehlt von 1 bis 100 und ersetzt dabei jedes Vielfache  
 * von fuenf durch das Wort "Fitz", jedes Vielfache von sieben  
 * durch das Wort "Quack" und jedes Vielfache von fuenf und sieben  
 * durch das Wort "FitzQuack".  
 * Das Programm benutzt den Modulo-Operator (%), um zu bestimmen,  
 * ob eine Zahl durch eine andere teilbar ist.  
 */  
  
public class FitzQuack {  
    public static void main(String[] args) {  
        for(int i = 1; i <= 100; i++) {  
            if (((i % 5) == 0) && ((i % 7) == 0)) // von 1 bis 100 zaehlen  
                System.out.print("FitzQuack");  
            else if ((i % 5) == 0)  
                System.out.print("Fitz");  
            else if ((i % 7) == 0)  
                System.out.print("Quack");  
            else  
                System.out.print(i);  
            System.out.print(" ");  
        }  
        System.out.println();  
    }  
}
```

### Aufgabe 3

Geben Sie das folgende Programm im Editor ein und speichern Sie es in einer Datei namens Reverse.java ab (z. B. mit dem Kommando notepad Reverse.java). Übersetzen Sie das Programm mit dem Java-Compiler javac und lassen Sie es im Java-Interpretierer java ablaufen.

```
/**  
 * Das folgende Programm gibt die Kommandozeilen-Argumente  
 * in umgekehrter Reihenfolge und die Zeichen jedes Arguments  
 * wiederum rückwärts aus.  
 * Bsp.:  
 *      Aufruf des Programms:  java Reverse Otto ist brav.  
 *      Ausgabe des Programms: .varb tsi ottO  
 */  
  
public class Reverse {  
    public static void main(String[] args) {  
        // rückwärts das Argument-Array durchlaufen  
        for(int i = args.length - 1; i >= 0; i--) {  
            // rückwärts alle Zeichen jedes Arguments durchlaufen  
            for(int j = args[i].length()-1; j >= 0; j--) {  
                // Zeichen j von Argument i ausgeben  
                System.out.print(args[i].charAt(j));  
            }  
            // nach jedem Argument ein Leerzeichen ausgeben  
            System.out.print(" ");  
        }  
        // und am Ende die Zeile abschliessen  
        System.out.println();  
    }  
}
```

### Aufgabe 4

Kopieren Sie die Datei

JavaDoc.java aus uebung1.zip

in eines Ihrer Dateiverzeichnisse. Öffnen Sie die Datei im Editor notepad. Geben Sie das Kommando javadoc JavaDoc.java ein (nicht im Editor, sondern auf der Kommando-Ebene). Es werden einige HTML-Dateien erzeugt (nachschauen z. B. mit dem Kommando dir). Öffnen Sie darin die Datei JavaDoc.html (z. B. mit Doppelklick). Es wird der Netscape Communicator gestartet (oder der MicroSoft Internet Explorer). Vergleichen Sie die Ausgabe von netscape mit dem Quelltext der Datei JavaDoc.java.

### Aufgabe 5

Das Dateiverzeichnis

uebung1.zip

enthält weitere Java-Quelldateien. Kopieren Sie diese Dateien in eines Ihrer Dateiverzeichnisse. Übersetzen Sie die Dateien mit dem Java-Compiler javac und lassen Sie die Programme im Java-Interpretierer java ablaufen.