

## **Teil**

# **Imperative Programmierung**

## **Unterrichtseinheit 12**

## **Java-Programme und -Entwicklungswerkzeuge**

**Dr. Dietrich Boles**

- Installation
- Programmerstellung
- Unterschiede zum Hamster-Modell
- Ein- und Ausgabe
- Beispiele
- Eclipse
- Zusammenfassung

- <http://java.sun.com/>
- Download J2SE <X> Update <Y> JDK (nicht JRE!)
- Installieren (Installationsverzeichnis merken!)
- Unter Umständen die PATH-Umgebungsvariable erweitern um  
    <Installationsverzeichnis>\bin
- In dem Verzeichnis befinden sich die Befehle, um die Standard-Java-Entwicklungswerkzeuge (Compiler **javac**, Interpreter **java**) aufzurufen

(1) Erstellen des Programm-Quellcodes (Editor):

```
class World {  
    public static void main(String[] args) {  
Anweisungen → |        System.out.println("Hello world!");  
                    }  
                }  
                    }
```

↑  
String-Literal

Demo

(2) Abspeichern des Quellcodes in einer Datei mit dem Namen  
**<Klassenbezeichner>.java**

Speichern in Datei: **World.java**

(3) Compilieren des Quellcodes (Compiler `javac`)

hier: `javac World.java`

(4) Solange der Compiler Fehlermeldungen liefert

(4.1) Fehler beseitigen (Editor)

(4.2) erneut compilieren (Compiler)

(5) Findet der Compiler keine Fehler, erzeugt er eine Datei, die Java-Bytecode enthält; sie trägt den Namen

`<Klassenbezeichner>.class`

hier: `World.class`

(6) Ausführung des Programms (mittels Interpreter `java`)

hier: `java World`

- alle Java-Programme sind Klassen

```
class <Programmname> {  
  
    <Funktionen und globale Variablen>  
  
    public static void main(String[] args) {  
        <Anweisungen>  
    }  
}
```

- vor Prozeduren und Funktionen: **static**
- vor globalen Variablen: **static**
- Hamster-Befehle existieren (natürlich) nicht

➤ **Beispielaufgabe:**

Berechnung des Quadrats einer bestimmten Zahl

➤ **Programm (1):**

```
class Quadratzahl1 {  
    public static void main(String[] args) {  
        int zahl = 8;  
        int quadratZahl = zahl * zahl;  
        System.out.println(quadratZahl);  
    }  
}
```

Demo

## ➤ Programm (2):

```
class Quadratzahl2 {  
  
    static int quadratZahl = 0;  
  
    static void berechnen() {  
        int zahl = 8;  
        quadratZahl = zahl * zahl;  
    }  
  
    public static void main(String[] args) {  
        berechnen();  
        System.out.println(quadratZahl);  
    }  
}
```

Demo



- Vordefinierte Java-Klasse `IO`
- Dateien:
  - `IO.java` (Java-Quellcode)
  - `IO.class` (Java-Bytecode)
  - `IO.README` (Informationen)
- Die Datei `IO.class` muss sich in demselben Verzeichnis befinden, wie ein Java-Programm, das compiliert bzw. ausgeführt werden soll !
- Eingabeanweisungen; z.B. `int zahl = IO.readInt();`
- Ausgabeanweisungen;
  - Java-Standard: `System.out.println("hello!");`
  - in `IO`, z.B.: `IO.println("hello!");`

```
class Ausgaben {  
    public static void main(String[] args) {  
  
        System.out.print(4711);  
        System.out.println(" ist eine positive Zahl!");  
        int fuenf = 5;  
        IO.print(-fuenf);  
        IO.println(" ist eine negative Zahl!");  
  
    }  
}
```

Demo

```
class Eingaben {  
    public static void main(String[] args) {  
  
        int zahl = IO.readInt("Zahl: ");  
        int quadratZahl = zahl * zahl;  
        System.out.print("Quadratzahl: ");  
        System.out.println(quadratZahl);  
        System.out.print("Kubikzahl: ");  
        System.out.println(quadratZahl * zahl);  
  
    }  
}
```

Demo

Programm zur Flächenberechnung eines Rechtecks:

```
class RechteckFlaeche {  
    public static void main(String[] args) {  
  
        System.out.println(  
            "Berechnung der Flaeche eines Rechtecks");  
        int laenge = IO.readInt("Laenge: ");  
        int breite = IO.readInt("Breite: ");  
        System.out.print("Flaeche = ");  
        System.out.println(laenge * breite);  
  
    }  
}
```

Demo

Schreiben Sie ein Programm "*Umkehr*", das zunächst eine Zahl von der Tastatur einliest und diese dann in umgekehrter Reihenfolge wieder ausgibt.

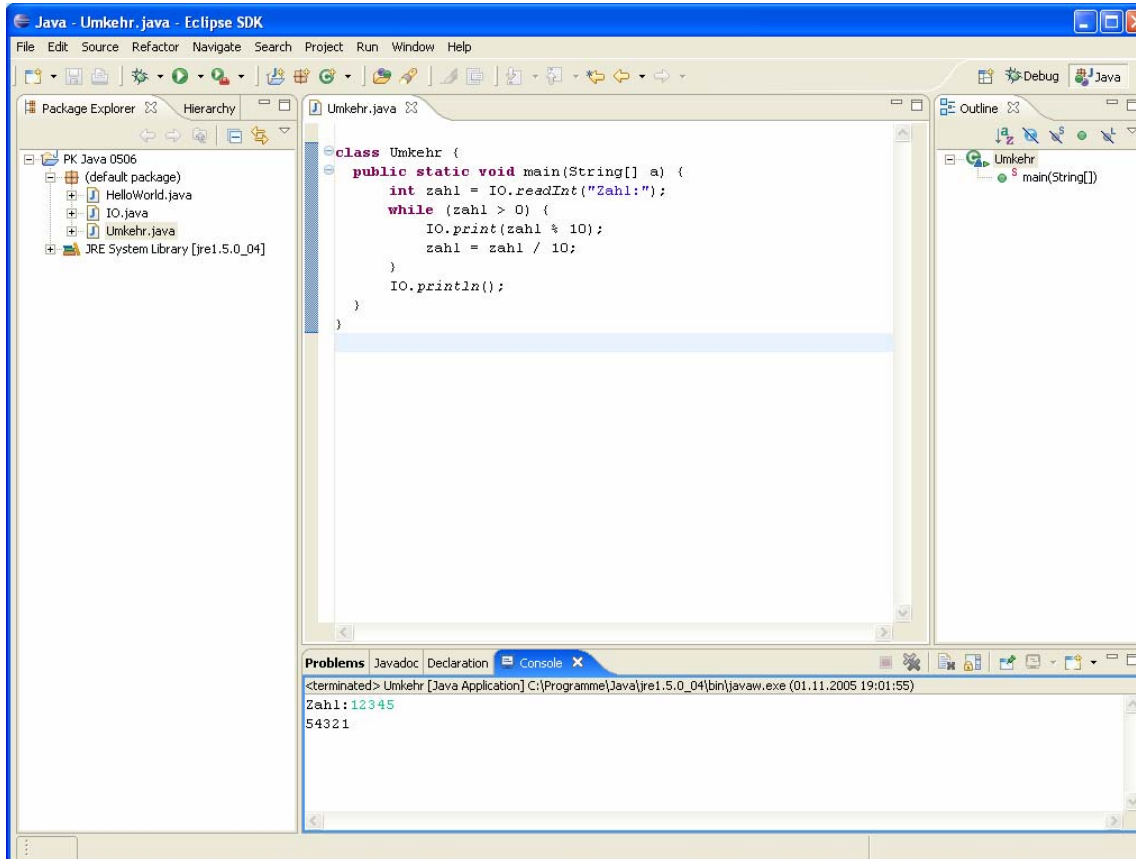
Beispiel:

```
$ java Umkehr  
Zahl:  
3567<CR>  
7653  
$ java Umkehr  
Zahl:  
1210<CR>  
0121  
$
```

```
class Umkehr {  
    public static void main(String[] a) {  
        int zahl = IO.readInt("Zahl:");  
        while (zahl > 0) {  
            IO.print(zahl % 10);  
            zahl = zahl / 10;  
        }  
        IO.println();  
    }  
}
```

Demo

- komplexe, mächtige Java-Entwicklungsumgebung
- [www.eclipse.org](http://www.eclipse.org)



Demo

- Java SE JDK enthält alle benötigten Werkzeuge zum Entwickeln und Ausführen von Java-Programmen
- Empfehlung: spezielle Entwicklungsumgebungen benutzen
  
- Java-Programme
  - Klassen
  - `public-static-void-main`-Funktion
  
- Ein-/Ausgabe:
  - Klasse IO benutzen