

# Einstieg in die Informatik mit Java

## Streams

Gerd Bohlender

Institut für Angewandte und Numerische Mathematik

- 1 Einführung
- 2 Klassenhierarchie (Byteorientiert)
- 3 Klassenhierarchie (Zeichenorientiert)
- 4 Auftretende Probleme

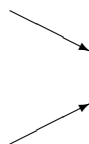
- 1 Einführung
- 2 Klassenhierarchie (Byteorientiert)
- 3 Klassenhierarchie (Zeichenorientiert)
- 4 Auftretende Probleme

Die Ein- und Ausgabe erfolgt in Java über *Ströme* bzw. (*Streams*).

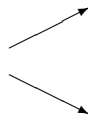
Diese laufen immer nur in eine Richtung:

Eingabestrom 1  
z.B. Tastatur

Eingabestrom 2  
z.B. Datei



Programm

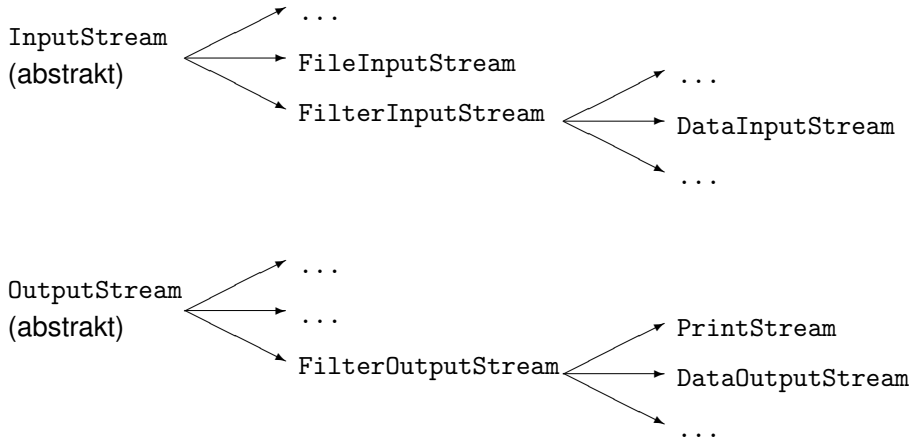


Ausgabestrom 1  
z.B. Bildschirm

Ausgabestrom 2  
z.B. Datei

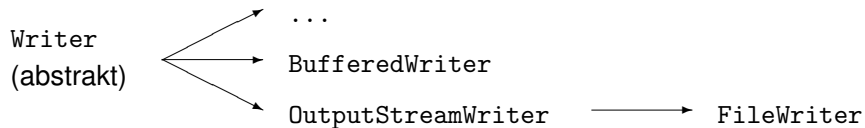
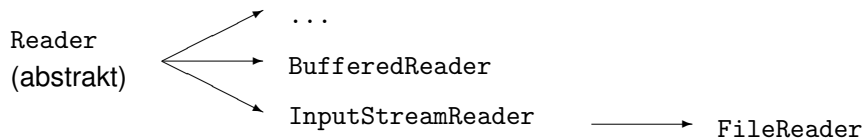
- 1 Einführung
- 2 Klassenhierarchie (Byteorientiert)**
- 3 Klassenhierarchie (Zeichenorientiert)
- 4 Auftretende Probleme

# Klassenhierarchie (Byteorientiert)



- 1 Einführung
- 2 Klassenhierarchie (Byteorientiert)
- 3 Klassenhierarchie (Zeichenorientiert)**
- 4 Auftretende Probleme

# Klassenhierarchie (Zeichenorientiert)





- 1 Einführung
- 2 Klassenhierarchie (Byteorientiert)
- 3 Klassenhierarchie (Zeichenorientiert)
- 4 Auftretende Probleme**

# Auftretende Probleme

- (1) Java verwendet den Unicode Zeichensatz (16 Bit), dieser wird allerdings von manchen Betriebssystemen nicht verstanden. Daher muss zuvor eine Wandlung in den ASCII Zeichensatz (8 Bit) vorgenommen werden.
- (2) Die Übertragung mehrerer Zeichen über einen Puffer ist i.allg. effizienter (Netzwerk, Festplatte, usw.). Daher erfolgt eine Pufferung der Daten in `BufferedReader` und `BufferedWriter`.
- (3) Die Zeilenenden sind abhängig vom Betriebssystem. Windows verwendet dabei `\r`, `\n`, Unix hingegen nur `\n`. Die Klassen `BufferedReader` und `BufferedWriter` stellen für diesen Zweck die Methoden `newLine()` und `readLine()` zur Verfügung.

Weitere Punkte auf der nächsten Seite

- (4) Wie erhält man die Ein- und Ausgabestreams auf die Standardein- und Standardausgabe?

```
System.in  InputStream // Standardeingabe  
System.out PrintStream // Standardausgabe  
System.err PrintStream // Fehlermeldungen
```

## Achtung

in, out und err sind statische Elemente der Klasse System.