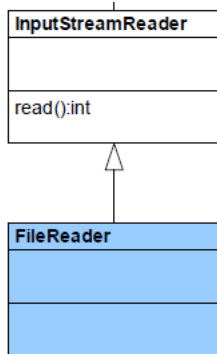


## Kontrollfragen A

1. Ein JavaProgramm liest Daten aus einem ???strom.  
*Input-Stream (Eingabestrom)*
  
2. Ein JavaProgramm schreibt Daten in einen ???strom.  
*Output-Stream (Ausgabestrom)*
  
3. Welche 2 Arten von Datenströmen unterscheidet Java? Wie sind die entsprechenden Klassen benannt?  
*Byte-Datenströme:*
  - *InputStream*
  - *OutputStream**Zeichen-Datenströme:*
  - *Reader*
  - *Writer*
  
4. Wie heissen die beiden Klassen, welche den I/O mit elementaren Datentypen unterstützen?
  - *DataInputStream*
  - *DataOutputStream*
  
5. Wie schaltet man bei Java programmiertechnisch Datenströme hintereinander?  
*Der Vorgänger wird dem Nachfolger als aktueller Parameter dem Konstruktor übergeben.*

```
FileInputStream aFileInputStream = new FileInputStream("data.bin");
DataInputStream aDataInputStream = new DataInputStream(aFileInputStream);
double value = aDataInputStream.readDouble();
```

6. In welcher Beziehung stehen die Klassen `FileReader` und `InputStreamReader`?



*FileReader erweitert InputStreamReader*

*Es sind zusätzliche Funktionen implementiert, welche das Datei-Handling vereinfachen.*

## Kontrollfragen B

### 1. Inwiefern unterscheiden sich binäre Dateien und Text Dateien?

*Binäre Dateien:*

- Folge von Bytes, beliebige Daten speicherbar

*Text Dateien:*

- Folge von Zeichen, File mit Editor lesbar

### 2. Was versteht man im Zusammenhang mit Dateien unter "Random Access"?

*„wahlfreier Zugriff“, es kann direkt jede beliebige Stelle in einer Datei gelesen werden (zugriff erfolgt nicht sequentiell)*

### 3. Wie öffnet man in Java eine Datei?

```
String fileName = "data2.bin";
File aFile = new File(fileName);
FileOutputStream aFileOutputStream = new FileOutputStream(aFile);
// DataOutputStream unterstützt elementare Datentypen
DataOutputStream aDataOutputStream = new DataOutputStream(aFileOutputStream);
```

### 4. Beim Lesen einer binären Datei liefert read() ein Byte bzw. einen Wert 0... 255 zurück. Wie wird EOF signalisiert?

*Mit dem Wert -1.*

### 5. Was bewirkt die Methode flush()?

*Puffer pro aktiv "spülen" / übertragen.*

⇒ *Wichtig bei Netzwerkverbindungen:*

- *Die Daten werden erst übertragen wenn flush() aufgerufen wird!*

## Kontrollfragen C

### 1. Wozu dient die Objekt-Serialisierung?

*Die ObjektSerialisierung ermöglicht es, Objekte mit Hilfe von Datenströmen übertragen zu können, z.B. in eine Datei oder via Internet auf einen anderen Rechner.*

*Bei der Serialisierung eines Objektes werden seine Attribute in einen ByteDatenstrom geschrieben (Klasse ObjectOutputStream) und umgekehrt aus einem ByteDatenstrom gelesen (Klasse ObjectInputStream).*

### 2. Was für Variablen werden nicht serialisiert?

- *Statische Attribute (Klassenvariablen)*
- *Transiente Instanzvariablen (Modifizier: transient)*
- *Konstanten*

### 3. Welches Interface muss eine Klasse implementieren, damit ihre Objekte serialisierbar sind?

*java.io.Serializable*

```
import java.io.Serializable;  
public class Matrix implements Serializable
```

### 4. Was ist das Besondere an diesem Interface?

*Das Interface besitzt keine abstrakten Methoden, somit müssen keine Methoden dadurch implementiert werden.*

*Das Interface ist nur ein „Marker“.*

### 5. Was für Objekte nennt man persistent?

*Objekte, die ausserhalb eines Laufzeitsystems existieren (ausserhalb JVM), werden persistente Objekte genannt.*

*Persistent: „nicht flüchtig“*

### 6. Das Serialisieren eines bestimmten Objektes kann zahlreiche weitere Objekt-Serialisierungen nach sich ziehen. Erklären Sie.

*Damit man ein Objekt aus einem Datenstrom wieder rekonstruieren kann, müssen oft auch alle vom Objekt direkt und indirekt referenzierten Objekte mit serialisiert werden (automatisches rekursives Prozedere)!*

*Es wird das aktuelle Objekt und alle Objekte welche darauf referenzieren serialisiert – resp. wieder eingelesen.*