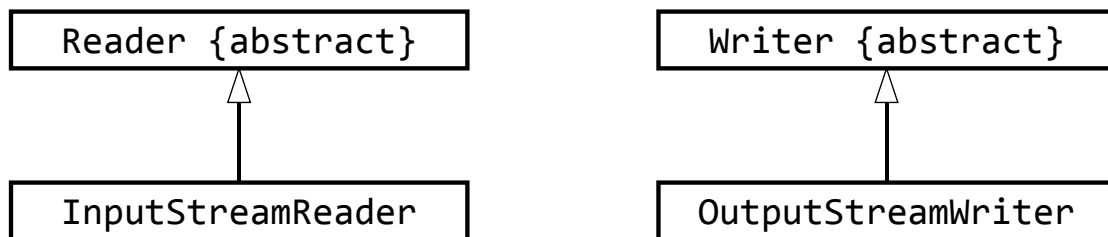


## Character-Streams

Während die Unterklassen von `OutputStream` und `InputStream` das Senden und Empfangen von Byte-Folgen realisieren, sind die Unterklassen von `Writer` und `Reader` für das Senden und Empfangen von Zeichenfolgen verantwortlich.



Dazu werden die Zeichenfolgen mit Hilfe einer Kodierung in Byte-Folgen verwandelt, dann als Byte-Stream übertragen und schließlich wieder in eine Zeichenfolge zurückverwandelt.

Wie Zeichen in Bytes kodiert werden, hängt nicht zuletzt vom Betriebssystem und den geografischen Einstellungen ab. Möchte man von der Standard-Kodierung abweichen, so kann bei der Instanziierung des `Writer` bzw. `Reader` auch eine Kodierungstabelle mit angegeben werden, z.B.:

```
FileOutputStream fos = new FileOutputStream("test.txt");
Writer out = new OutputStreamWriter(fos, "UTF8");
```

Welche Kodierungstabellen zur Verfügung stehen, entnehmen Sie bitte der Java API Dokumentation.

## Aufgabe „exercise10“ (Abgabe 20.06. bis 23.06.2016, bZv-relevant)

Bei textbasierten Anwendungen muss die Tastatureingabe häufig zeilenweise ausgewertet werden. Allerdings wird die Tastatureingabe in Java über `System.in` als Byte-Stream zur Verfügung gestellt, der zunächst keine Zeilenorientierung kennt. Gegenstand dieser Übung ist es, diese Byte-Folge in eine Zeichenfolge umzuwandeln.

Im Verzeichnis `exercise10` finden Sie das Interface `EchoFinderInterface` mit folgenden Methoden:

```
public abstract String getPair(BufferedReader bufRead);  
public abstract BufferedReader getSystemInAsBufferedReader();
```

### Ihre Aufgabe:

Erstellen Sie eine neue Klasse `EchoFinder`, die das `EchoFinderInterface` wie folgt implementiert:

- Die Methode `getPair` soll den übergebenen Character-Stream so lange zeilenweise lesen, bis entweder das Ende des Streams erreicht ist, oder zwei aufeinanderfolgende Zeilen den gleichen Inhalt besitzen. Wurde das Ende des Streams erreicht, so soll `null` zurückgegeben werden, ansonsten der Inhalt der zuletzt gelesenen Zeile, d.h. das „Echo“ der Zeile zuvor.
- Die Methode `getSystemInAsBufferedReader` soll den Byte-Stream der Tastatureingabe als Character-Stream zurückliefern.

Um Ihre Lösung zu überprüfen, müssen Sie außerdem in Ihrer Klasse `EchoFinder` eine `main`-Methode implementieren. Nach dem Start des Programms sollen so lange von der Tastatur zeilenweise Eingaben gelesen werden, bis zwei aufeinanderfolgende Zeilen den gleichen Inhalt besitzen. Zur Kontrolle soll dieser doppelte Inhalt auf der Konsole mittels `System.out.println` ausgegeben werden.

Die Implementierung der `main`-Methode ist bei dieser Übung Bestandteil der Aufgabe!