

Aufgabenblatt 1

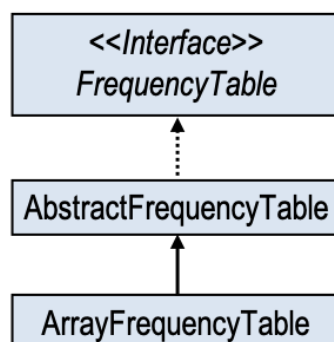
Es soll ein Typ `FrequencyTable` realisiert werden, mit dem eine Häufigkeitstabelle für Wörter verwaltet werden kann. Beispielsweise soll der Programmausschnitt

```
FrequencyTable ft = new ArrayFrequencyTable();  
ft.add("die");  
ft.add("das");  
ft.add("die");  
System.out.println(ft);
```

alle Wörter mit ihren Häufigkeiten ausgeben:

```
{die:2, das:1}
```

Gegeben ist dazu das Interface `FrequencyTable`, das Sie ebenso wie die im folgenden beschriebenen Java-Klassen in dem Archiv `src.zip` finden. In dem Archiv `docs.zip` finden Sie die Javadoc-Dokumentation zum Interface `FrequencyTable` und der Hilfsklasse `Word`. In der Klasse `ArrayFrequencyTable` soll eine Häufigkeitstabelle als Feld realisiert werden. Dabei wird eine abstrakte Klasse `AbstractFrequencyTable` als Zwischenschritt eingebaut. (Der Zweck der abstrakten Klasse wird erst im Aufgabenblatt 2 ersichtlich.)

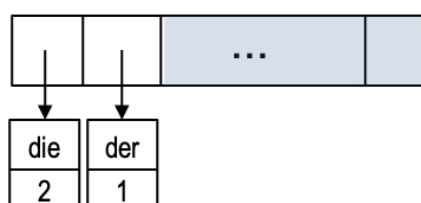


Klasse **AbstractFrequencyTable**

Von dieser Klasse ist eine rudimentäre Implementierung gegeben, die Sie vervollständigen sollen.

Klasse **ArrayFrequencyTable**

Realisieren Sie eine Häufigkeitstabelle als Feld. Das Feld enthält alle Wort-Häufigkeits-Paare nach den Häufigkeiten absteigend sortiert. Die Abbildung zeigt das Feld für das Beispiel von oben.



Auch diese Klasse ist teilweise vorgegeben und soll von Ihnen vervollständigt werden.

Testen mit der Klasse `ArrayFrequencyTable_Test`

Außerdem vorgegeben ist eine Testklasse mit zwei Testszenarien:

1. Es werden kleine Häufigkeitstabellen erstellt und alle Methoden getestet. Prüfen Sie, ob ihre Implementierung zum gewünschten Ergebnis führt.
2. In der Datei `Kafka_Der_Prozess.txt` finden Sie einen Roman als Textdatei. Führen Sie eine Häufigkeitsanalyse durch. Im Testprogramm ist bereits eine Schleife vorhanden, die alle Wörter eines deutschsprachigen Textes einliest und ausgibt (Leerraum-, Satzzeichen und andere Sonderzeichen werden als Trennzeichen interpretiert). Ergänzen Sie das Testprogramm geeignet.