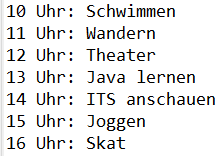
**Ziel: mittel**

Schreiben Sie ein Programm „Terminkalender“, dass Ihnen die Uhrzeit sowie den anstehenden Termin ausgibt. Verwenden Sie ein Array mit Namen „termin“ sowie eine FOR-Schleife.

Die Ausgabe sollte in etwa so aussehen:



**Ziel: sehr schwer**

Es sollen die Metainformationen eines MP3-Files ausgelesen werden. Diese findet man im so genannten ID3-Tag ([engl.](https://de.wikipedia.org/wiki/Englische_Sprache) identify an MP3“). Es bezeichnet ein Format für Zusatzinformationen (Metadaten), die in Audiodateien des MP3-Formats enthalten sein können.

ID3v1 ist einfach gehalten und besteht einzig aus einem 128 [Bytes](https://de.wikipedia.org/wiki/Byte) großen Block mit festem Aufbau:

|  |  |  |
| --- | --- | --- |
| Offset | Länge | Bedeutung |
| 0 | 3 | Kennung „TAG“ zur Kennzeichnung eines ID3v1-Blocks |
| 3 | 30 | Titel des Musikstücks |
| 33 | 30 | Künstler/Interpret |
| 63 | 30 | Album |
| 93 | 4 | Erscheinungsjahr |
| 97 | 30 | Beliebiger Kommentar |
| 127 | 1 | Genre |

Das Genre hat eine Länge von 1 Byte. Die Bedeutung der Zahl können Sie folgender Liste entnehmen:



**Hinweise:**

Erzeugen Sie sich eine Klasse „ID3Tagauslesen“.

Importieren Sie aus „java.io“ die benötigten Pakete (FileInputStream; FileNotFoundException; IOException).

Die Datei „Song6.mp3“ soll direkt im Verzeichnis D: liegen. Erzeugen Sie eine Instanz "fis" (*fileinputstream*) der Klasse FileInputStream.

FileInputStream fis = **new** FileInputStream("d:/Song6.mp3");

Deklarieren Sie die einzelnen Abschnitte des ID3-Tags entsprechend der obigen Tabelle also: tag, titel, interpret, cdtitel, jahr, kommentar, genre. Legen Sie dazu die erforderlichen Byte-Arrays mit den in der Tabelle angegebenen Größen an. Für genre dürfen Sie kein Byte Array verwenden, da es sich lediglich um eine Zahl handelt.

Da Sie außerdem die Größe der Datei ausgeben wollen, müssen Sie sich noch eine Variable groesseInByte anlegen. Überlegen Sie sich genau, welchen Datentyp Sie für diese Variable nehmen. Es kann durchaus MP3-Dateien geben, die größer als 3 GB sind.

Um die Größe zu bestimmen benötigen Sie die Methode available(). Sie wird auf die Instanz „fis“ angewendet. Lassen Sie sich als Erstes lediglich die Größe der Datei in Bytes ausgeben.

groesseInByte = fis.available();

// Datensatzzeiger positionieren

fis.skip(groesseInByte - 128);

System.***out***.println("No. of bytes remaining to read after skipping: " + fis.available());

Da der ID3-Tag immer am Ende der MP3-Datei steht, ist es relativ einfach, ihn auszulesen. Mittels der Methode skip() können Sie eine bestimmte Anzahl an Bytes überspringen und den Datensatzzeiger an die gewünschte Stelle positionieren. Lassen Sie sich nach dem „Sprungbefehl“ die Anzahl der verbleibenden Bytes ausgeben.

Lesen Sie dann die Arrays für das Tag (Größe 3 Bytes) sowie den Titel (Größe 30 Bytes aus) und lassen Sie sich abermals die Anzahl der verbleibenden Bytes ausgeben.

Lesen Sie anschließend die weiteren Felder ein und geben Sie diese dann aus. Dazu müssen Sie die Byte-Arrays in String Variablen umwandeln. Beispiel:

fis.read(tag);

String strTAG = new String(tag);

Bei genre müssen Sie an das explizite Casting denken:

genre = (byte) fis.read();

**Übrigens:**

Einer der besten Tag-Editoren, mit dem man schnell und einfach die Titel-Informationen (ID3-Tags) von Musik-Dateien erstellen oder verändern kann und den es gratis zum Download gibt, ist MP3tag.

**Ausgabe/Darstellung:**

