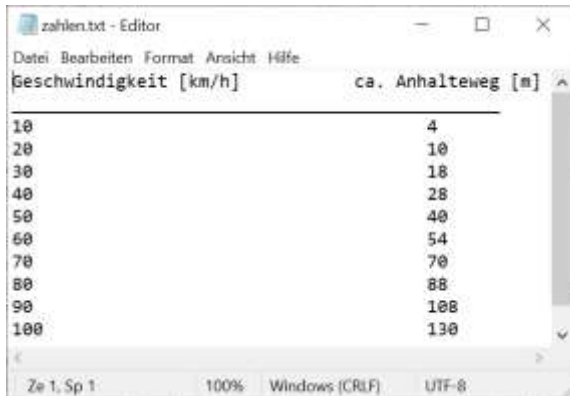


Durchführung

Arbeitsblatt- Schreiben von Textdateien mit Java APIs

1. Aufgabe:- Anhalteweg Formel aus der Fahrschule

Schreiben Sie ein Programm, dass Daten (Anhalteweg-Formel Berechnung) formatiert in einer Datei schreibt. Benutzen Sie das vorgegebene Dateiformat.



Geschwindigkeit [km/h]	ca. Anhalteweg [m]
10	4
20	10
30	18
40	28
50	40
60	54
70	70
80	88
90	108
100	130

Die ersten beiden Dateizeilen sind Zeichenketten. Mittels einer **for**-Schleife sollen die restlichen Dateizeilen in die Datei geschrieben werden. Als Trennzeichen zwischen den Spaltenwerten benutzen Sie einen Tabulator »**␣**«. Das Schreiben in einer Textdatei sollte immer mit geeigneter Fehlerbehandlung (**try-catch**-Konstrukt) abgesichert werden.

Verwenden Sie die Methode *Files.write* und schreiben Sie eine Liste in die Datei mit Namen **zahlen.txt**

Zum Berechnen des Bremsweges soll eine statische Methode `calcBremsweg()` aufgerufen werden und die komplette Liste (mit allen Datenzeilen) von dieser statischen Methode zurückgegeben werden.

Es sollen folgende Faustformeln benutzt werden:

Normaler Bremsweg (in m) $\approx (\text{Geschwindigkeit} \div 10) \times (\text{Geschwindigkeit} \div 10)$

Reaktionsweg (in m) $\approx (\text{Geschwindigkeit} \div 10) \times 3$

Bremsweg + Reaktionsweg = Anhalteweg

2. Aufgabe

Gegeben ist die Datei **gradcelsius.txt**. Schreiben Sie ein Programm, das die Temperaturdaten (°C) aus der Datei **gradcelsius.txt** liest, die Temperaturdaten in Kelvin (K) umrechnet und am Bildschirm ausgibt. Die Temperaturdaten in °C und in K sollen formatiert, getrennt durch ein Semikolon und am Ende mit einem Zeilenumbruch, in eine Datei geschrieben werden.

Das Schreiben in einer Textdatei sollte immer mit geeigneter Fehlerbehandlung **try-catch**-Konstrukt abgesichert werden.

3. Aufgabe

Schreiben Sie ein elementares Kopierprogramm, dass eine bereits vorhandene Datei in eine neue Datei kopiert. Das Programm soll Daten aus einer vorhandenen Quelldatei lesen und in eine Zieldatei kopieren/schreiben. Der Benutzer soll über Tastatur die Dateinamen für Quelldatei und Zieldatei eingeben. Das Lesen/Schreiben aus/in einer Textdatei sollte immer mit geeigneter Fehlerbehandlung **try-catch-finally**-Konstrukt abgesichert werden.

Im finally Konstrukt kann z.B. der Scanner zugemacht werden.

Lesen Sie die Datei in einer **while**-Schleife bitte Zeichen für Zeichen ein und schreiben Sie jedes eingelesene Zeichen gleich in die Ausgabedatei.

Als Hilfe soll Ihnen der Code-Abschnitt dienen, diesen können Sie so modifizieren, dass sie die eingelesenen Zeichen sofort wieder „herausschreiben“. Recherchieren Sie nach den notwendigen Klassen im Netz.

```
import java.io.FileReader;
import java.io.IOException;

public class A3_CopyforSign {

    public static void main(String[] args) {
        FileReader fr = null;
        try
        {
            String fileName="C:\\tmp\\zahlen.txt";
            fr = new FileReader(fileName);
            StringBuffer sb = new StringBuffer();
            int ch;
            while( (ch=fr.read()) != -1 )
                sb.append((char)ch);
            // Hier wird nur auf dem Bildschirm gedruckt
            System.out.println(sb.toString());
        }
        catch(IOException ex)
        {
            ex.printStackTrace();
        }
        finally
        {
            if(fr!=null)
                try {
                    fr.close();
                } catch (IOException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
        }
    }
}
```