

### Aufgabe: Exam (JUnit Tests, Streams, Fluent Interface)

Durch die bZv-Vorbereitung kennen Sie sich bereits gut mit Multiple Choice (MC) Tests aus. In dieser Aufgabe sollen Sie folgende Klasse (Exam) erweitern, die einen Test aus Textfragen repräsentiert.

```
public class Exam {  
    //Liest den Tex-Header aus einer Datei u. gibt ihn als String zurück  
    String readHeaderFromFile() {...}  
}
```

Die Klasse soll das Einlesen von Fragen aus einer Datei und die Erzeugung eines Tests im LaTeX-Format ermöglichen. In den Teilaufgaben sind die Anforderungen an die Exam Klasse genannt, die Sie implementieren sollen.

Gegeben sei ferner die Klasse Question:

```
public class Question {  
  
    String text; //Fragentext  
  
    //Wandelt eine eingelesene Zeile in ein Question-Objekt um  
    public static Question toQuestion(String line) { ... }  
}
```

- a) Die Klasse Exam soll mehrere Question-Objekte haben können. In der lokalen Datei *questions.csv* sind zeilenweise Fragen gespeichert. Ergänzen Sie die Klasse Exam um eine Methode *readQuestions*,
- die die Datei *questions.csv* zeilenweise ausliest,
  - jede eingelesene Zeile an die Methode *toQuestion* der Klasse Question übergibt
  - jedes resultierende Question-Objekt im Exam speichert
  - die die Propagierung einer IOException angibt.
- b) Wenn das Exam fertig konfiguriert ist, soll es als Text in eine Datei geschrieben werden. Ergänzen Sie die Klasse Exam um eine Methode *toTest*, die
- zunächst den mit der Methode *readHeaderFromFile* gelesenen Inhalt in die Datei *test.tex* schreibt,
  - die Question-Objekte im Exam in folgendes Format überführt:

`\textbf{Fragentext}\\`

Beispiel für die Frage **Was besagt das Substitutionsprinzip?** :

`\textbf{Was besagt das Substitutionsprinzip?}\\`

und auch in die Datei *test.tex* (hinter den Header) zeilenweise schreibt,

- die Datei mit dem String `\end{document}` abschließt,
- die Propagierung einer `IOException` angibt.

**Hinweis: Zur Illustration:** Eine Tex-Datei sieht bspw. wie folgt aus:

```
\documentclass[10pt,a4paper]{article}
\usepackage[margin=2.5cm, rmargin=2.5cm]{geometry}
\usepackage[utf8]{inputenc}
...
\begin{document}

\textbf{Was besagt das Substitutionsprinzip?}\\
...

\end{document}
```

- c) Schreiben Sie einen Junit Test, der beide Methoden in der richtigen Reihenfolge aufruft. Der Test soll erfolgreich durchlaufen, wenn die Datei *test.tex* existiert.

**Hinweis:** Wenn Sie versuchen, eine Datei, die es nicht gibt, mit einem Stream zu öffnen, wirft die JVM eine `FileNotFoundException`.

- d) Skizzieren Sie kurz mit Hilfe von Java Code, wie Sie mit Hilfe eines Fluent Interfaces sicherstellen können, dass ein Entwickler, die Methodenaufrufe `readQuestions` und `toTest` nur in dieser Reihenfolge absetzen kann. Sie müssen nicht nochmals die Methoden aus den Teilaufgaben a) und b) implementieren.