

PLF 18.4.2024

Klassen: 1AHWII / 4AAIF

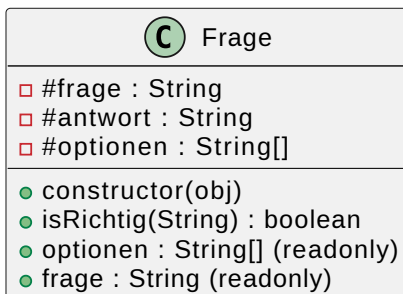
Themen: Javascript Basics if/else, Schleifen, Klassen

Vorbemerkung

1. Verschieben Sie alles vom Laufwerk `Z:\` auf den Desktop
2. Entpacken Sie die Angabe-Datei
3. Öffnen Sie diese mit dem Windows-Explorer
4. Öffnen Sie den `PLF` Workspace der sich darin befindet.
5. stellen Sie mittels `npm run` oder `package.json` fest, dass Sie öfters `npm start` bzw. `npm test` aufrufen werden
6. Bearbeiten Sie die Datei `plf.js`.
7. Lösen Sie die Aufgaben
8. Schließen Sie vscode
9. Verschieben Sie den Angabe-Ordner wieder auf Laufwerk `Z:\`.

Aufgabe 1: gegeben ist folgendes UML Diagramm

... für eine Quiz-Anwendung:



In der Datei `fragen.js` wird ein JSON Objekt exportiert, welches ein Array aus Fragen beinhaltet. Jedes dieser Objekte kann als Parameter für den konstruktor obiger `Frage`-Klasse verwendet werden.

Programmieren Sie die Datei `plf.js` und beachten Sie die dortigen Kommentare!

Implementieren Sie den constructor derart, dass das übergebene Objekt die privaten Variablen befüllt.

`optionen` soll als property implementiert werden. Beim Lesen dieses properties soll das private `optionen`-Objekt zurückgegeben werden, beim Schreiben dieses properties soll eine exception geworfen werden.

`frage` soll als reine readonly property implementiert werden. Ein Schreiben des `frage` properties möge "schweigend" nichts tun.

Letztlich soll die Methode `isRichtig(String)` `true` zurückliefern, wenn der übergebene String ident mit der Antwort ist, andernfalls `false`.

In `package.json` unter "scripts" findet sich "start" und "test" zum Ausführen, mit und ohne debugging Modus.

Aufgabe 2: Berechnen der Euler'schen Zahl e

Die Berechnung von e ist wie folgt definiert:

$$e = 1 + \frac{1}{1} + \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3} + \frac{1}{1 \cdot 2 \cdot 3 \cdot 4} + \dots = \sum_{k=0}^{\infty} \frac{1}{k!}$$

Machen Sie aus `eulerZahl` eine `function`, welche in 20 Iterationen die Zahl berechnet und zurückgibt.

Gutes Gelingen