Prof. Dr. Stephan Kleuker
Hochschule Osnabrück
Fakultät Ing-Wiss. und Informatik
- Software-Entwicklung -

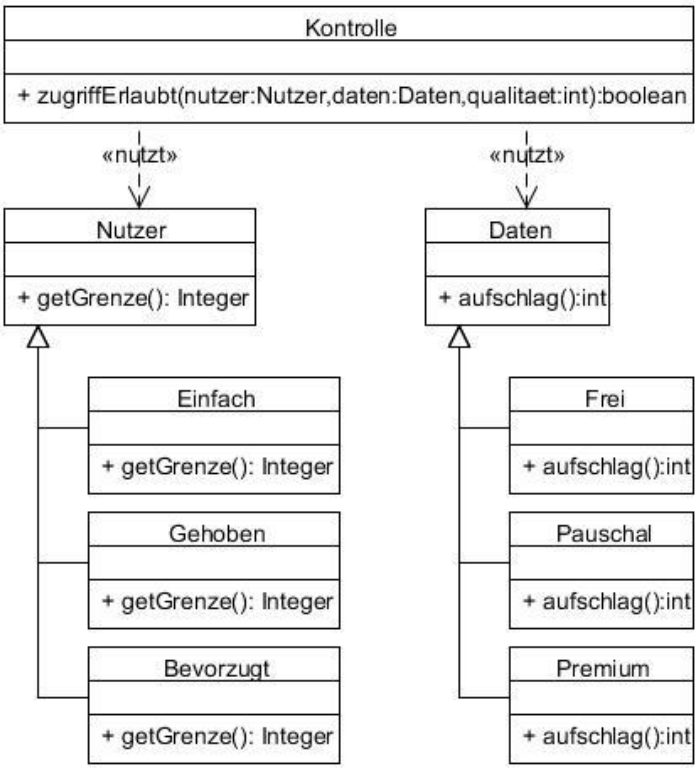
Software-Qualität

Sommersemester 2025
4. Aufgabenblatt

Aufgabe 0.4 (1 Punkt)
Geben Sie das Lösungswort des Quiz aus der Lernnotiz an.

Aufgabe 7 (2 Punkte)
Diese Aufgabe setzt die vorherige Aufgabe 4 fort, laden Sie zwei der dort hochgeladenen Testklassen herunter. Welche Ergebnisse liefern die Tests für Ihre Implementierung, nachdem Sie Ihre eigene Klasse in die Tests eingebaut haben? Mussten Sie Ihre Implementierung anpassen, um die anderen Tests nutzen zu können, wenn ja dokumentieren Sie, warum.

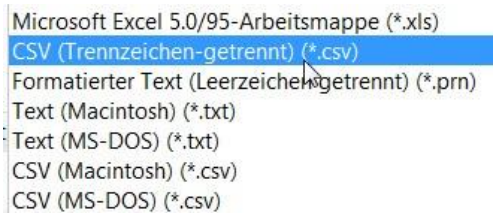
Aufgabe 8 (6 Punkte)
Spezifikation: Zu entwickeln ist eine Kontrollklasse Kontrolle mit einer Booleschen Methode zugriffErlaubt, die als Parameter ein Nutzerobjekt, ein Datenobjekt und einen ganzzahligen Qualitätswert erhält. Es gibt die Nutzerklassen „Einfach“, „Gehoben“ und „Bevorzugt“, die Datenklassen „Frei“, „Pauschal“ und „Premium“. Die Methode soll eine Ausnahme werfen, falls das Nutzerobjekt von der Klasse „Einfach“ oder die Qualität über 100 ist. Die Methode liefert für ein Nutzerobjekt der Klasse „Gehoben“ genau dann wahr, wenn das Datenobjekt zu „Frei“ gehört und der Qualitätswert unter 60 liegt oder das Datenobjekt zu „Pauschal“ gehört und der Qualitätswert unter 40 liegt. Die Methode liefert für ein Nutzerobjekt der Klasse „Bevorzugt“ genau dann wahr, wenn das Datenobjekt zu „Frei“ mit einem beliebigen erlaubten Qualitätswert oder „Pauschal“ gehört und der Qualitätswert unter 80 liegt oder das Datenobjekt zu „Premium“ gehört und der Qualitätswert unter 60 liegt.



Auf der rechten Seite befindet sich ein zugehöriger Ausschnitt aus einem Klassendiagramm, das auf die wesentlichen Informationen reduziert ist, wobei es genau nur um die Methode zugriffErlaubt(.) geht.

a) Gegeben sei obige informelle Aufgabenspezifikation und die von der Veranstaltungswebseite erhältliche Implementierung in einem Projekt qsAufgabeAequivalenz. Überlegen Sie sich anhand der Äquivalenzklassenmethode zusammen mit der Grenzwertanalyse systematisch Testfälle und setzen Sie diese Testfälle mit JUnit um. Beachten Sie, dass bei der reinen Äquivalenzklassenanalyse keine Zusammenhänge zwischen Parametern beachtet werden. (Eventuell hat die vorliegende Implementierung sogar Fehler).

b) Im Ordner src befindet sich ein Excel-Sheet, das zur Eingabe von Testdaten und erwarteten Ergebnissen genutzt werden kann. Nach dem

	A	B	C	D
1	**Nutzer**	**Datum**	**Qualität**	**Ergebnis**
2	Gehoben	Frei	50	true
3	Gehoben	Pauschal	30	true
4	Bevorzugt	Frei	60	true
5	Bevorzugt	Pauschal	70	true
6	Bevorzugt	Premium	30	true
7	Bevorzugt	Premium	80	false
8	Bevorzugt	Premium	120	Fehler
9	Gehoben	Premium	10	false
10	Einfach	Premium	130	Fehler

Export als csv-Datei (siehe rechts, eventuell muss vorher das gewählte Trennzeichen auf Komma gesetzt werden oder jedes Semikolon ist durch ein Komma zu ersetzen), befinden sich die Daten in der Datei Testdaten.csv. Schreiben Sie eine weitere JUnit-Testklasse, die die Daten aus der Datei einliest und die zugehörigen Tests parametrisiert durchführt.