

Aufgabe 1 (10 Punkte)

- a) Benenne die 4 Varianztypen bei der generischen Programmierung (2 Punkte)
- b) Erläutere kurz den wesentlichen Zweck von inneren Klassen (2 Punkte)
- c) Erläutere kurz den wesentlichen Unterschied zwischen intermediären und terminalen Operationen (2 Punkte)
- d) Skizziere die Testpyramide (2 Punkte)
- e) Erläutere kurz, was man unter der Ereignisbehandlung versteht (2 Punkte)

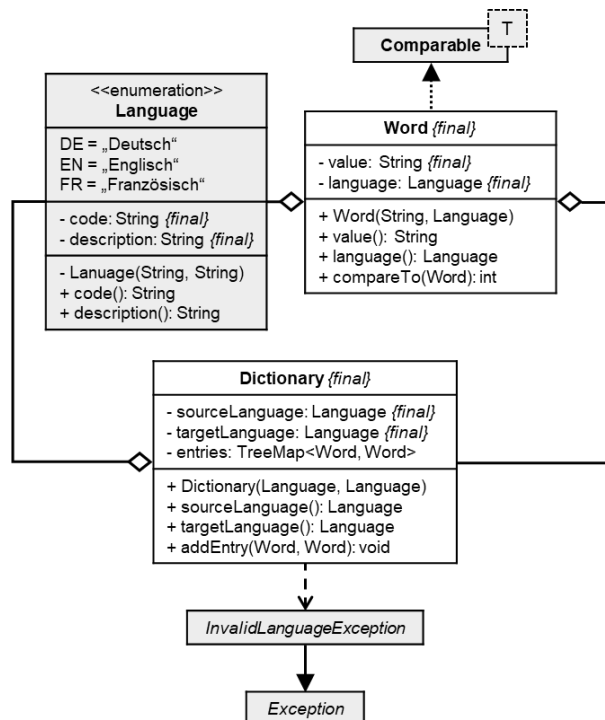
Themengebiete

Generische Programmierung, Innere Klassen, Die Java Stream API, Softwaretests, JavaFX-Anwendungen

Aufgabe 2 (10 Punkte)

Erstelle die Klassen Word (3 Punkte) und Dictionary (7 Punkte) anhand des abgebildeten Klassendiagramms.

Klassendiagramm



Hinweise zur Klasse Word

- Der Konstruktor soll alle Attribute initialisieren
- Die Methoden `value()` und `language()` sollen den Wert des entsprechenden Attributs zurückgeben
- Die Methode `compareTo(Word)` soll so implementiert werden, dass damit Wörter aufsteigend nach ihrem Wert (`value`) sortiert werden können

Hinweise zur Klasse Dictionary

- Der Konstruktor soll alle Attribute initialisieren
- Die Methoden `sourceLanguage()` und `targetLanguage()` sollen den Wert des entsprechenden Attributs zurückgeben
- Die Methode `addEntry(Word, Word)` soll dem Wörterbuch die beiden eingehenden Wörter hinzufügen. Für den Fall, dass die Sprache des ersten eingehenden Wortes (`language`) nicht der Quellsprache des Wörterbuchs (`sourceLanguage`) entspricht, oder dass die Sprache des zweiten eingehenden Wortes (`language`) nicht der Zielsprache des Wörterbuchs (`targetLanguage`) entspricht, soll die Ausnahme `InvalidLanguageException` ausgelöst werden.

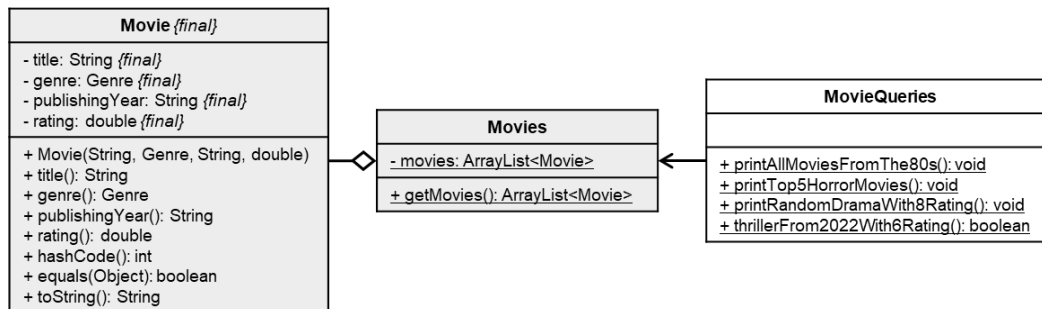
Themengebiete

Abstrakte und finale Klassen und Methoden, Schnittstellen, Komparatoren, Ausnahmen

Aufgabe 3 (12 Punkte)

Erstelle die Klasse MovieQueries anhand des abgebildeten Klassendiagramms.

Klassendiagramm



Hinweise zur Klasse MovieQueries

- Die statische Methode `printAllMoviesFromThe80s()` soll alle Filme der Klasse `Movies` zwischen 1980 und 1990 in der Form *FILMTITEL (Erscheinungsjahr)* ausgeben
- Die statische Methode `printTop5HorrorMovies()` soll die ersten 5 Horror-Filme der Klasse `Movies` absteigend sortiert nach der Bewertung ausgeben
- Die statische Methode `printRandomDramaWith8Rating()` soll eine beliebiges Drama der Klasse `Movies` mit einer Bewertung besser als 8 ausgeben
- Die statische Methode `thrillerFrom2022With6Rating()` soll zurückgeben, ob es in der Klasse `Movies` einen Thriller aus dem Jahr 2022 mit einer Bewertung von 5,5 bis 6,4 gibt

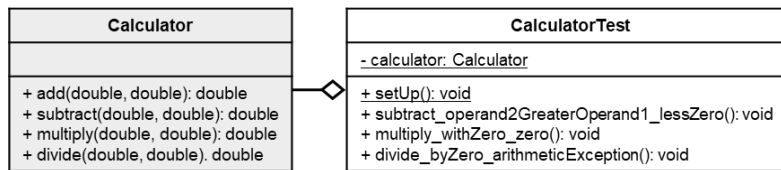
Themengebiete

Schnittstellen, Komparatoren, Generische Programmierung, Innere Klassen, Die Java Stream API

Aufgabe 4 (8 Punkte)

Erstelle die Testklasse `CalculatorTest` anhand des abgebildeten Klassendiagramms.

Klassendiagramm



Hinweise zur Testklasse `CalculatorTest`

- Die statische Lebenszyklus-Methode `setUp()` soll sicherstellen, dass vor dem Test alle Attribute der Klasse initialisiert werden
- Die Testmethode `subtract_operand2GreaterOperand1_lessZero()` soll den nachfolgenden Testfall abdecken:
 - Zu testende Methode: `subtract(double, double)`
 - Eingabe: die zweite eingehende Zahl ist größer als die erste eingehende Zahl
 - Erwartetes Ergebnis: Zahl kleiner Null
- Die Testmethode `multiply_withZero_zero()` soll den nachfolgenden Testfall abdecken:
 - Zu testende Methode: `multiply(double, double)`
 - Eingabe: mindestens eine der beiden eingehenden Zahlen ist Null
 - Erwartetes Ergebnis: Null
- Die Testmethode `divide_byZero_arithmeticException()` soll den nachfolgenden Testfall abdecken:
 - Zu testende Methode: `divide(double, double)`
 - Eingabe: die zweite eingehende Zahl ist Null
 - Erwartetes Ergebnis: `ArithmeticException`

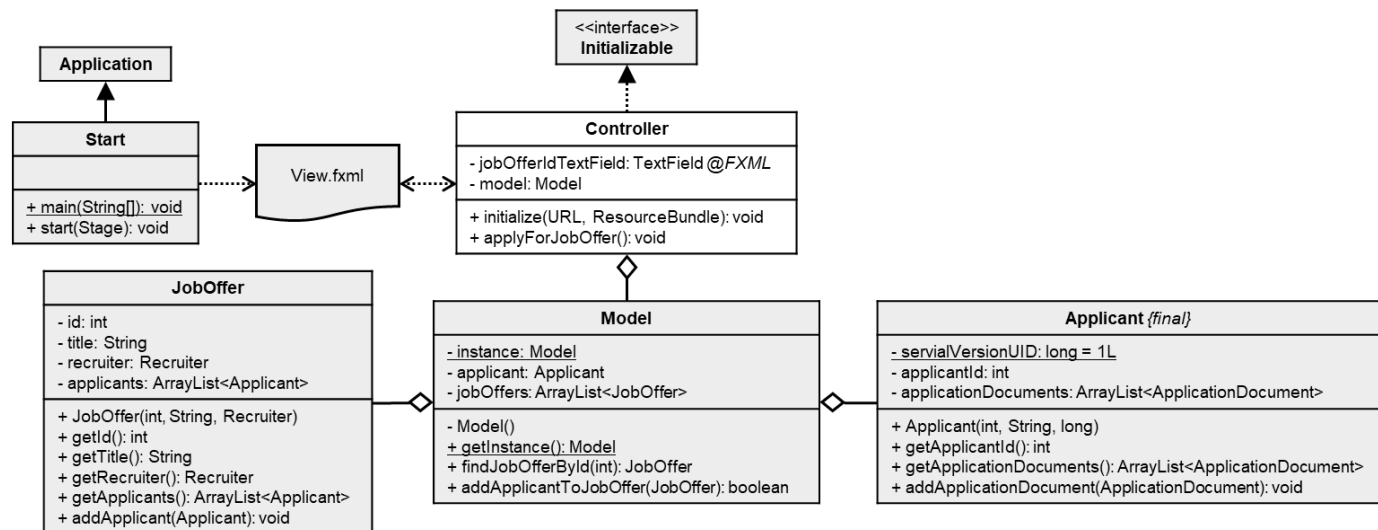
Themengebiete

Komponententests

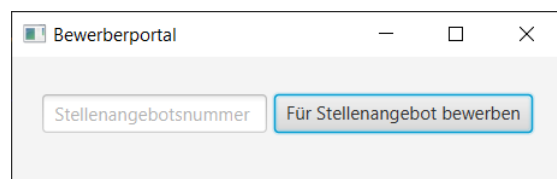
Aufgabe 5 (10 Punkte)

Erstelle die Klasse Controller anhand des abgebildeten Klassendiagramms.

Klassendiagramm



Grafische Darstellung



Hinweise zur Klasse Controller

- Die Methode `initialize(URL, ResourceBundle)` soll das Model initialisieren
- Die Methode `applyForJobOffer()` soll zur eingegebenen Stellenangebotsnummer das dazugehörige Stellenangebot suchen (Methode `findJobOfferById(int)`) und diesem anschließend den Bewerber hinzufügen (Methode `addApplicantToJobOffer(JobOffer)`). Für den Fall, dass es sich bei der Eingabe um keine ganze Zahl handelt, soll ein Dialog (`AlertType.ERROR`) mit der Fehlermeldung „Bitte eine Ganzzahl eingeben“ ausgegeben werden und für den Fall, dass zur eingegebenen Stellenangebotsnummer kein dazugehöriges Stellenangebot gefunden werden konnte, soll ein Dialog (`AlertType.ERROR`) mit der Fehlermeldung „Bitte eine gültige Stellenangebotsnummer eingeben“ ausgegeben werden. In beiden Fällen soll die Verarbeitung der Methode abgebrochen werden.

Hinweis

Die Methode `valueOf(String)` der Klasse `Integer` kann die Ausnahme `NumberFormatException` auslösen.

Themengebiete

Schnittstellen, JavaFX-Anwendungen