

Lernziele

Eigene Klassen erstellen und andere Klassen benutzen können.

In einer Klasse Termin soll ein Zeitpunkt (Jahr, Monat, Tag) und eine Beschreibung dazu gespeichert werden. Hierfür muss die Klasse Termin zwei Attribute besitzen, sodass der Zeitpunkt und die Beschreibung geeignet gespeichert werden können. Die Zeitpunkte verschiedener Termine sollen miteinander verglichen werden können. Diese Funktion soll die Klasse Termin bereitstellen, sodass jedes Objekt einer Klasse Termin überprüfen kann, ob ein anderes Termin-Objekt

1. zu einem früheren
2. zum selben oder
3. zu einem späteren Zeitpunkt

stattfindet.

Eine Klasse TerminplanerUI soll mehrere Termindaten einlesen und sortiert nach dem Datum speichern. Es dürfen nur zukünftige Termine eingegeben werden. Auf Anforderung sollen alle im Terminplaner enthaltenen Termine aufgelistet werden. Standardmäßig ist im Terminplaner ein persönlicher Termin von Ihnen enthalten, der in der Beschreibung Ihren Namen und Vornamen speichert, dieser wird nicht über die Konsole eingelesen.

Aufgabenstellung

Beispiel:

Es wird über die Konsole ein Treffen mit Hans eingelesen. Das Treffen findet am 15.12.2020 statt. Der Termin wird im Terminplaner gespeichert. Anschließend wird ein weiterer Termin eingelesen. Findet der zweite Termin zu einem früheren Zeitpunkt statt als das Treffen mit Hans, wird der Termin vor diesem im Kalender gespeichert. Wenn er zu einem späteren Zeitpunkt stattfindet dahinter. Wie die Ein-/Ausgaben auf der Konsole aussehen sollen, finden Sie beim Lösungshinweis 1.

(a) Recherche (max. 15 Punkte)

Suchen Sie in der Java-API nach einer geeigneten Klasse, mit deren Hilfe Sie Datumswerte speichern können, und geben Sie diese an.

Beschreiben Sie mit eigenen Worten, welche Operationen Sie wie verwenden möchten.

Hinweis: Wenn Sie (a) nicht lösen können, überlegen Sie sich bitte eine eigene geeignete Datenstruktur zur Speicherung des Datums und verwenden diese für die folgenden Teilaufgaben.

(b) UML-Klassendiagramm (max. 15 Punkte)

Zeichnen Sie die Klassen Termin und TerminplanerUI in ein UML-Klassendiagramm. Führen Sie alle Attribute und Operationen mit Angabe von Typen auf, die Ihre Klassen besitzen.

(c) Klasse Termin (max. 10 Punkte)

Programmieren Sie die Klasse Termin, die den oben aufgeführten Anforderungen genügt. Überlegen Sie sich eine eigene geeignete Datenstruktur zur Speicherung des Datums. Dokumentieren Sie jede Operation mit Kommentaren.

(d) UI-Klasse (max. 60 Punkte)

Programmieren Sie die Klasse TerminplanerUI, die den oben aufgeführten Anforderungen genügt und die folgende Funktionalitäten aufweist:

/1/ Beim Start wird auf der Konsole folgendes Menü angezeigt:

```
-----  
Bitte wählen Sie zwischen folgenden Optionen:  
Neuer Termin: 1  
Termine anzeigen: 2  
Anwendung beenden: 3  
-----
```

Gibt der Benutzer eine 1 ein, folgt /2/.

Gibt er eine 2 ein, folgt /3/.

Gibt er eine 3 ein, wird der Terminplaner beendet.

Gibt der Benutzer einen anderen Wert ein, soll ein Hinweis auf der Konsole erscheinen und erneut das Menü angezeigt werden.

/2/ Der Benutzer hat eine 1 eingegeben, nun wird er zur Eingabe seiner Termindaten aufgefordert (Datum, Beschreibung).

Gibt der Benutzer ungültige Werte ein, soll ein Hinweis auf der Konsole erscheinen und erneut das Menü aus /1/ angezeigt werden. Sind die Werte korrekt wird ein neuer Termin angelegt und in einer geeigneten Datenstruktur in der Klasse TerminplanerUI, nach Datum sortiert, gespeichert.

/3/ Der Benutzer hat eine 2 eingegeben, nun werden alle bereits eingegebenen Termine in der Reihenfolge ausgegeben, wie Sie im Terminplaner gespeichert wurden.

Dokumentieren Sie jede Operation mit Kommentaren. Laden Sie das UML-Diagramm als jpg- oder png-Datei, sowie die .java Dateien einschl. der Ausgaben eines Testdurchlaufes als Text-Datei zusammen in einer .rar oder .zip Datei auf den Server.

Hinweise:

- Verwenden Sie zum Einlesen das Paket inout.
- Die Datei des UML Diagramms muss Ihren Namen und das Erstellungsdatum enthalten.
- Fehlt der fest einprogrammierte, persönliche Termin so wird die gesamte Aufgabe mit 0 Punkten bewertet!