

Übung 02: JavaFX

Abgabetermin: 28. 3. 2022, 8:30

Übung 2: Task Kalender

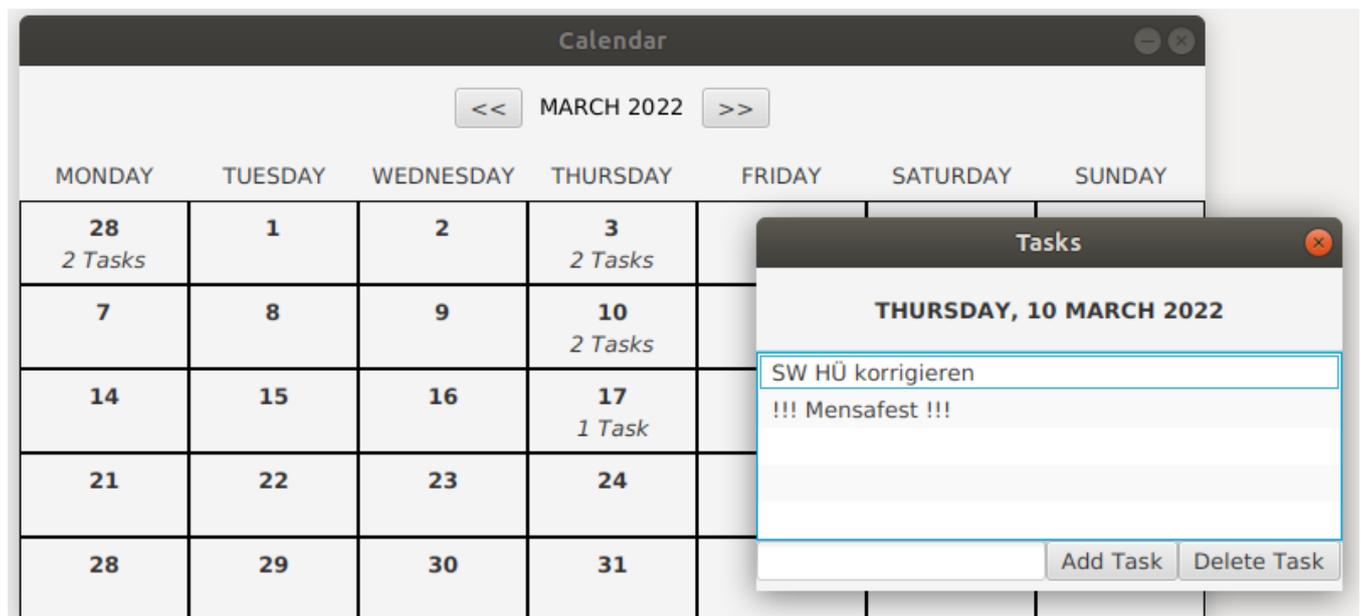
(24 Punkte)

Ihr *JayUnit* Framework hat sich mittlerweile bei den Instituten herumgesprochen, wodurch sie sich nicht mehr vor Tutorenjobs retten können. Um bei den vielen Korrekturdeadlines den Überblick nicht zu verlieren, entschließen Sie sich eine einfache Kalenderapplikation zum Verwalten von Aufgaben (Tasks) zu erstellen

Folgende Aktionen sollen im Kalender möglich sein:

- Anzeigen des aktuellen Monats
- Kalenderansicht, in der die Anzahl der Tasks für jeden Tag angezeigt wird
- Erstellen, Bearbeiten und Löschen von Tasks

Verwenden Sie ausschließlich JavaFX zur Implementierung der GUI. Verwenden Sie außerdem bei der Implementierung JavaFX-Features (Properties, FXML, ...) wo angebracht. Die folgende Abbildung zeigt eine Beispielimplementierung der beschriebenen GUI:



Datenmodell

Sie bekommen das Datenmodell bestehend aus den Klassen `CalendarModel`, `Day` und `Task` zur Verfügung gestellt.

- `CalendarModel` berechnet selbstständig alle Tage des Monats beginnend mit einem Montag (eventuell des Vormonats) und endend mit einem Sonntag (eventuell des Folgemonats)
- Sämtliche Klassen können (und müssen teilweise noch) verändert werden!

Vorgehen und Anforderungen

- Der Einfachheit halber kann jeder „Monat“ mit 35 Tagen angenommen werden, was das Layout der KalenderGUI statisch macht. Über das CalendarModel können Sie sequentiell auf alle Tage des „Monats“ (inklusive der Füller-Tage) zugreifen.
- Jeder Tag in der Kalenderübersicht soll die Anzahl der Tasks anzeigen.
- Das Einsehen der Tasks soll über ein Pop-Up Fenster geschehen, welches bei einem Klick/Doppelklick auf einen Tag im Kalender geöffnet wird.
- Über dieses Pop-Up Fenster sollen Einträge bearbeitet, angefügt und gelöscht werden können.
 - Für das Bearbeiten können Sie eine ListView mit editierbaren TextFieldListCells verwenden.
 - Es sollen keine leeren Tasks angefügt werden können.
- Das CalendarModel speichert nur einen Monat, wodurch beim Wechseln des Monats sämtliche Tage im Model neu berechnet werden müssen und alle Tasks verloren gehen. Keine Panik! Das ist gewollt und wird in der nächsten Übung behoben.
- Achten Sie auf eine saubere Verwendung des MVC Prinzips. Änderungen in der GUI sollen Änderungen im Modell nach sich ziehen und umgekehrt.
 - Ergänzen Sie hierfür das zur Verfügung gestellten Datenmodell um JavaFX spezifische Komponenten, wie Properties oder Bindings.

Abgabe

Abzugeben ist das gesamte Projekt, direkt aus der IDE als .zip exportiert (Eclipse: Rechtsklick auf das Projekt → Export... → General → Archive File → als .zip speichern). Die .class Files brauchen nicht abgegeben werden (kann im Exportmenü abgewählt werden).

Hinweise:

- Sie können die GUI sowohl mittels FXML als auch programmatisch erstellen.
- Es empfiehlt sich, eine Hilfsklasse zu erstellen, um die GUI-Elemente für einzelne Kalendertage zu kapseln. Ein GridPane kann dann mit Objekten dieser Hilfsklasse befüllt werden.
- Verwenden Sie Bindings und Properties, um Änderungen im Model (Anzahl der Tasks bzw. Taskbeschreibung) in die GUI nachzuziehen.
- Zum Erzeugen des Add Pop-Ups müssen Sie eine neue Stage anlegen, der Sie per `initOwner` die Stage der Hauptapplikation übergeben. Ansonsten kann wie mit einer Main-Applikation verfahren werden (`FXMLLoader`, `Scene` setzen, ...)

Hinweis zu Übung 3:

In der nächsten Übung wird auf dieser Übung aufgebaut und ein Datenbankanschluss realisiert.