

SuperEasyShare

Synopsis

Implementieren Sie ausgewählte Teile eines Car Sharing Systems.

Projekt Import und Abgabe

- Holen Sie sich die zip Datei mit dem Projekt von Moodle
- Entzippen Sie die Datei
- Erstellen Sie ein neues Java Projekt (z.B. `MIDTERM_2023`) in IntelliJ
- Kopieren Sie den Ordner SuperEasyShare (in `src`) in den `src` Ordner in IntelliJ
- Öffnen Sie die `index.html` Datei in `apidocs` für die Java Dokumentation
- Zur Abgabe exportieren Sie das Projekt als Zip (File>Export) und geben es über Moodle ab

Einleitung

Für den Car-Sharing-Anbieter "SuperEasyShare" soll eine Software zur Verwaltung erstellt werden. In dem System werden sowohl die Stammdaten der Kunden als auch die verfügbaren Fahrzeuge und Leihvorgänge (intern als 'Trip' bezeichnet) erfasst und verwaltet. Der Lebenszyklus eines Trips beginnt mit der Erstellung, etwa durch Reservierung eines bestimmten Fahrzeugs durch einen bestimmten Kunden und einem bestimmten Tarif. Danach wird ein Trip zu einem bestimmten Zeitpunkt gestartet und schlussendlich zu einer bestimmten Zeit beendet. All diese Daten werden für jeden Trip erfasst.

Das Tarifmodell soll flexibel und vor allem erweiterbar gestaltet sein. Zur Zeit gibt es einen Minutentarif, bei dem die Gebühr von der Dauer des Trips abhängt und einen Langzeittarif, der einen gewissen Sockelbetrag für eine vereinbarte Leihdauer sowie eine zusätzliche Gebühr pro Tag Überschreitung vorsieht. Die aktive Marketingabteilung betont, dass sie ständig neue Tarife entwickelt und hier größtmögliche Flexibilität fordert, das System muss also leicht um neue Tarife ergänzt werden können.

Betrachten Sie das Klassendiagramm (1) um einen Überblick zu erlangen.

Aufgabe

Implementieren Sie die Klassen laut Spezifikation in den Javadocs.

Beachten Sie, dass im package `SuperEasyShare.provided` einige Klassen bereits gegeben sind und die Klasse `Main` Methoden zur Erstellung von Demodaten enthält.

Aufgabe	Punkte
Trip	18
- constructors	3
- get/set	3
- start	3
- end	3
- total	3
- toString	3
Rate	3
LongTermRate	3
PerMinuteRate	3
Main	3
- main	1
- print	2
total	30

Hinweise

- Implementieren Sie zunächst `Rate` und `Trip`.
- die Testdaten in `Main` funktionieren erst wenn die entsprechenden Klassen implementiert wurden, bis dahin können sie auskommentiert werden um Compile Errors zu vermeiden.

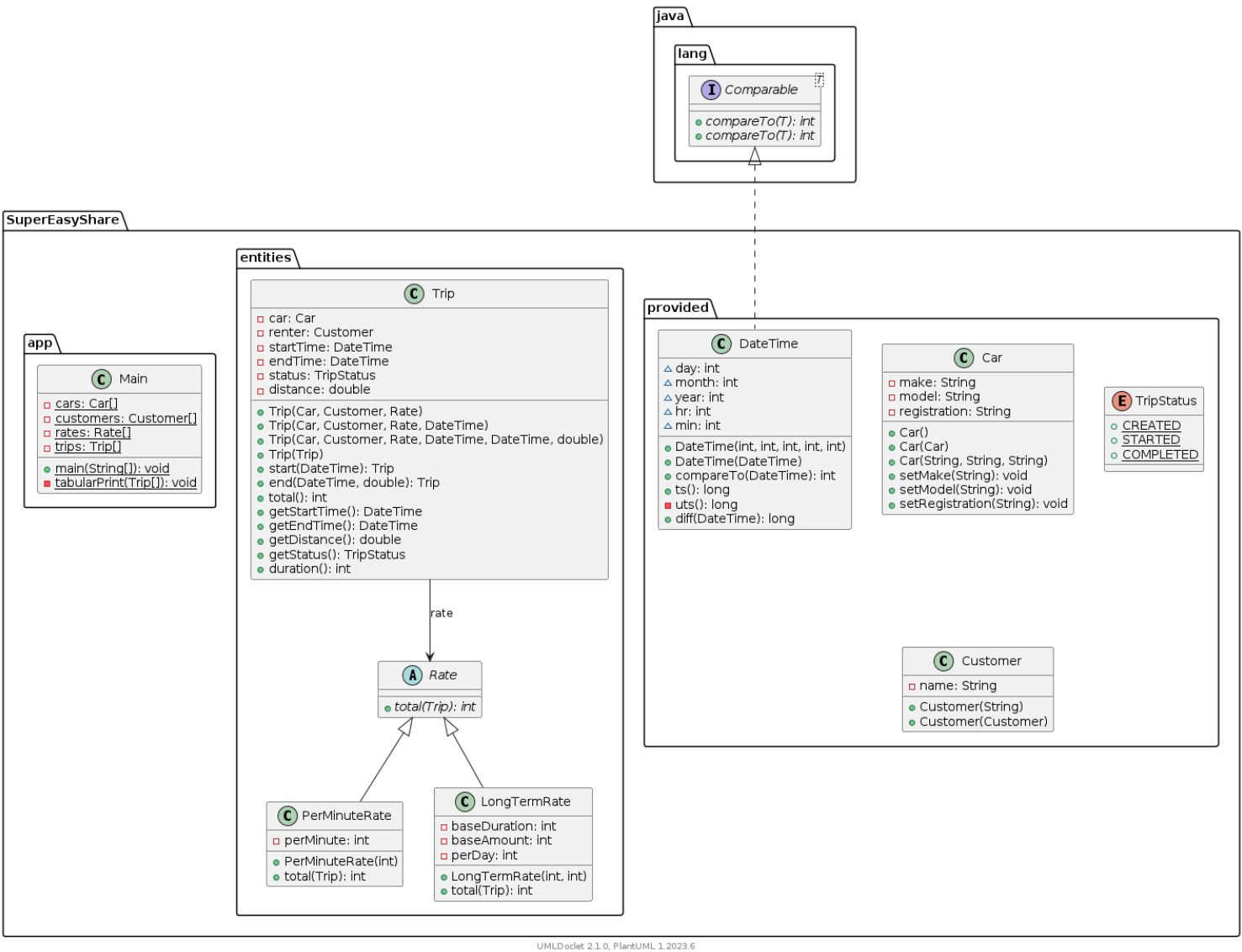


Figure 1 : Klassendiagramm SuperEasyShare