

Die Aufgaben müssen einzeln bearbeitet und abgegeben werden. Die Abgabe muss bis **spätestens Donnerstag 07.02.2012 um 23:59 Uhr** über unser Hausaufgabenabgabesystem <http://seblog.cs.uni-kassel.de/pmws1213/> erfolgen. Die Abgabe ist nur als einzelne *.zip oder *.jar-Datei möglich. Daher müssen alle für eine Abgabe relevanten Daten zu einem solchen Archiv kombiniert werden. Diese Hausaufgabe gibt **22 Punkte**.

Hinweise zur Abgabe:

- Die Hausaufgabe als exportiertes Eclipse Projekt (*.zip, **nicht** den gesamten Workspace) abgeben. Das kann mit Hilfe der Eclipse Export Funktion durchgeführt werden. Ist das Projekt nicht korrekt exportiert, kann es bei der Korrektur nicht berücksichtigt werden (es bietet sich also an, den Import des exportierten Projektes auszuprobieren).

WICHTIG Benennen Sie ihr Projekt für diese Abgabe nach folgendem Schema:

PMWS1213_HA<a>_<Matrikelnummer>,

wobei <a> für die aktuelle Hausaufgabe steht. Beispiel:

PMWS1213_HA10_12345678.

Allgemeines

Orientieren Sie sich für die Lösung der Aufgaben an den zugehörigen Übungen und Vorlesungen: <http://seblog.cs.uni-kassel.de/category/currentterm/pm-ws1213/>

Aufgabe 1: Projekt anlegen (2P)

Legen Sie ein neues Eclipse Projekt an und erstellen Sie eine SDMLib Class Model Klasse namens **RailwayClassModel.java**, die über eine main-Methode verfügt.

- Das Eclipse Projekt muss dem oben genannten Namensschema folgen:
PMWS1213_HA10_12345678
- Die SDMLib Class Model Klasse soll sich im src-Ordner des Projekts im Package "de.uks.se.railway" befinden.

Aufgabe 2: Textuelles Szenario (2P)

Erstellen Sie ein englisches, textuelles Szenario zu einem Zug, welcher Fahrgäste an mehreren aufeinander folgenden Haltestellen einsammelt. und speichern Sie es in `<Project Ordner>/doc/scenario.txt`.

Aufgabe 3: Objektdiagramme (4P)

Erstellen Sie zwei Objektdiagramme zu ihrem Szenario aus Aufgabe 2 in einem Werkzeug ihrer Wahl (nicht Fujaba!).

- Erstellen Sie ein Objektdiagramm zur Startsituation und speichern Sie es in `<Project Ordner>/doc/start.png`.
- Erstellen Sie ein Objektdiagramm zur Endsituation und speichern Sie es in `<Project Ordner>/doc/end.png`.

Aufgabe 4: Klassendiagramm (4P)

Leiten Sie aus den beiden in Aufgabe 3 erstellten Objektdiagrammen ein Klassendiagramm ab. Modellieren sie dieses in der main-Methode der SDMLib Class Model Klasse aus Aufgabe 1.

- Der Quellcode soll in den Ordner `<Project Ordner>/genSrc` generiert werden.
- Fügen Sie anschließend den Ordner `<Project Ordner>/genSrc` dem Build Path als `src` Ordner hinzu um etwaige Compilefehler angezeigt zu bekommen.

Aufgabe 5: SDMLib JUnit Test (6P)

Fügen Sie Modellcode aus Aufgabe 4 der Klasse `Train` eine Methode `pickUpAllPassengers()` hinzu. Erstellen Sie passend zu Aufgabe 3 ein SDMLib Scenario zum Einsammeln von Fahrgästen an mehreren Haltestellen.

- Legen Sie eine SDMLib Scenario Klasse (JUnit Test) im Package `de.uks.se.railway.tests` Ordner.
- Beheben Sie etwaige Compilefehler durch Hinzufügen der JUnit Bibliothek zum Projekt und/oder der Klasse (Quickfix nutzen).

Aufgabe 6:SDMLib Model Transformations (4P)

Erstellen Sie, wie in der Übung gezeigt, für die Methode `pickUpAllPassengers()` SDMLib Modell Transformationen im SDMLib Scenario aus Aufgabe 5, welche die Anforderungen des Tests aus Aufgabe 5 erfüllt (der Test muss erfolgreich durchlaufen, also »grün« werden). Extrahieren Sie abschließend die Modell Transformationen in die Methode `pickUpAllPassengers()` und ersetzen Sie die Modelltransformationen durch den Aufruf der Methode `pickUpAllPassengers()`.