



Die Aufgaben müssen von jedem Teilnehmer einzeln bearbeitet und abgegeben werden. Die Abgabe muss **bis spätestens Dienstag 10.06.2014 um 12:00 Uhr** über unser Hausaufgabenverwaltungssystem <https://se.cs.uni-kassel.de/hms/> 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.

Hinweise zur Abgabe:

- Die Hausaufgabe ist in Form von **zwei** exportierten Eclipse Projekten abzugeben. Mit Hilfe der Eclipse Export Funktion ist es möglich mehrere Projekte in eine zip-Datei zu exportieren. Ist ein 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 ihre Projekte für diese Abgabe nach folgendem Schema:

DPSS14_HA<a>__<Matrikelnummer>,

wobei <a> für die aktuelle Hausaufgabe und für die Aufgabennummer steht.

Beispiel für Aufgabe 1:

DPSS14_HA7_1_12345678.

Allgemeines

Orientieren Sie sich für die Lösung der Aufgaben an den zugehörigen Übungen und Vorlesungen: <http://seblog.cs.uni-kassel.de/category/currentterm/design-patterns2014/>

Die Benotung ergibt sich aus den Hausaufgaben, wobei eine Abgabe ausgelassen werden darf. Für die Note wird die nicht abgegebene Abgabe, beziehungsweise die Abgabe mit der geringsten Prozentzahl, nicht beachtet. Die Endnote ergibt sich aus dem Mittelwert der erreichten Prozentpunkte der übrigen Abgaben.

Die Veranstaltung gilt als nicht bestanden, wenn mehr als **eine Hausaufgabe nicht abgegeben** wurde oder der Mittelwert der zur Benotung herangezogenen Abgaben **weniger als 50%** beträgt.



Aufgabe 1 Factory Pattern

Implementieren Sie ein Datenmodell für eine Ausrüstungsfabrik für THW und Feuerwehr, welches in der Lage ist Schläuche und Pumpen herzustellen. Dabei ist das Factory Pattern zu verwenden.

- Bilden Sie dabei folgende Teile des Systems ab:
 - Hose
 - Pump
 - und die für das Pattern notwendige Klassen
- Schreiben Sie einen Junit-Test in welchem
 - eine sinnvolle Objektstruktur erzeugt wird (die Ausrüstungen gibt es in mindestens zwei Ausführungen):
 - hoses
 - pressure hoses (fire department)
 - suction hoses (technical emergency service)
 - Es müssen jeweils verschiedenen Maße (length/diameter) erstellt werden
 - pumps
 - bilge pumps (technical emergency service)
 - centrifugal pumps (fire department)
 - Auch hier sind jeweils mehrere Arten (flow rate) zu modellieren
 - sinnvolle Assertions verwendet werden



Aufgabe 2 Prototyp Pattern

Implementieren Sie ein Datenmodell für ein Gewächshaus unter Verwendung des Prototyp Pattern.

- Bilden Sie dabei folgende Teile des Systems ab:
 - Apple
 - Pear
 - Cherry
 - Plant
- Die Pflanzen besitzen verschiedene Eigenschaften: color, taste, shape, harvestTime, bugResistance, name
- Schreiben Sie einen Junit-Test in welchem
 - eine sinnvolle Objektstruktur erzeugt wird
 - jeweils der aktuelle Prototyp von jeder Pflanzen angelegt wird (hierzu sind verschiedene Belegungen der Attribute notwendig)
 - sinnvolle Assertions verwendet werden