



Die Aufgaben müssen einzeln bearbeitet und abgegeben werden. Die Abgabe muss **bis spätestens Montag, den 10.06.2013 um 23:59 Uhr** über unser Hausaufgabenabgabesystem <http://seblog.cs.uni-kassel.de/dpss13/> 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 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:

DPSS13_HA<a>_<Matrikelnummer>,

wobei <a> für die aktuelle Hausaufgabe steht.

Beispiel:

DPSS13_HA7_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-patterns/>

Die Benotung beruht auf den Hausaufgaben.

Hierfür werden die gesamten Hausaufgaben minus zwei vom Studenten abgegebenen Hausaufgaben addiert, die mit mehr als 50 % bewertet wurden. Nicht abgegebene Hausaufgaben oder Betrugsversuche bekommen 0 Punkte.

Die Prüfung gilt als nicht bestanden, wenn alle Hausaufgabenpunkte weniger als **50 %** ergeben oder der Student mehr als **zwei Hausaufgaben nicht abgibt**.



Aufgabe 1 (Abstract Factory) (3P)

- Hierfür soll eine Fahrzeugfabrik entworfen werden
- Die Erstellung von Fahrzeugen soll nach dem Abstract Factory Pattern erfolgen
- Es werden folgende Elemente benötigt: AbstractFactory, VWGolf, OpelVectra, VWConstellation, OpelBlitz und die für das Pattern erforderlichen Klassen
- Die Methoden createCar und createTruck sollen mittels der jeweiligen Konfiguration (VW bzw. Opel) entsprechende Instanzen der verschiedenen Fabrikate zurückliefern
- Das System soll in einem Test, der alle Instanzen erzeugt, getestet werden



Aufgabe 2 (Prototype-Pattern) (3P)

- In dieser Aufgabe soll ein Ausschnitt einer Software für Architekturbüros nach dem Prototype Pattern entwickelt werden
- Hierfür werden folgende Elemente benötigt: House und die für das Pattern benötigte Struktur
- Der Architekt hat eine Musterhaus mit 3 Fenstern, 1 Tür, 2 Etagen und ohne Keller entworfen
- Bei der Entwicklung soll eine sinnvolle Benennung der Attribute berücksichtigt werden
- Das System soll in einem Test an einem Haus mit 4 Etagen und 5 Fenstern mit Keller getestet werden