



Aufgaben für die vorlesungsfreie Zeit

Die Vorlesung kann nur einen ersten Einblick in die objektorientierte Programmierung liefern, der den ersten Schritt zur sinnvollen Software-Entwicklung bildet. Einige wichtige generelle Themen zu Java und zur Entwicklung generell werden teilweise auch in anderen Veranstaltungen nur kurz angesprochen, so dass hier ein selbstständiges Lernen von Studierenden erwartet wird. Dies unterstützen die folgenden Aufgaben.

Aufgabe 1: Java unterstützt beim Exception-Handling das sogenannte „try with resources“ bei dem ein try-Block wie folgt anfangen kann.

```
try (PrintWriter writer = new PrintWriter(new File("datei.txt"))) {  
    writer.println("Hello again");  
}
```

Finden Sie heraus, wie dieser Ansatz funktioniert und schreiben Sie ein eigenes Beispielprogramm dazu.

Aufgabe 2: Um Interfaces einfach um Methoden ergänzen zu können, erlaubt Java in Interfaces default-Implementierungen, wodurch es sich dann eher um eine abstrakte Klasse handelt. Informieren Sie sich über diesen Ansatz. Überlegen Sie zur Aufgabe mit der Erstellung von Rahmen und ausgefüllten Objekten, wo eine default-Implementierung sinnvoll sein könnte und setzen Sie es um.

Aufgabe 3: In der Veranstaltung wurden nur XML-Dateien zum Schreiben und Lesen von Objekten genutzt. Informieren Sie sich, wie man einfache Textdateien in Java lesen und schreiben kann. Implementieren Sie ein eigenes Beispiel.

Aufgabe 4: Neben List und Set gibt es in Java direkt Map als drittes zentrales Interface zur Programmierung von Sammlungen. Maps basieren auf dem Schlüssel-Wert-Prinzip, bei dem jedem Schlüssel ein Wert zugeordnet werden kann. Informieren Sie sich genauer, wie die Klasse `HashMap<,,>` funktioniert und programmieren Sie ein Beispiel. Zeigen Sie dabei, dass die Implementierung der Methode `hashCode()` wichtig ist.

Aufgabe 5: Die Software-Entwicklung findet meist in größeren Entwicklungsumgebungen statt. Arbeiten Sie sich soweit in die Entwicklungsumgebung Eclipse ein, so dass Sie mehrere Ihrer Praktikumslösungen dort zum Laufen bringen. Die folgende Anleitung könnte hilfreich sein: <http://home.edvsz.hs-osnabrueck.de/skleuker/querschnittlich/SEU.pdf>. Achten Sie darauf, dass sich Ihre Klassen in Paketen befinden, Sie sind bald keine Erstsemester mehr.

Aufgabe 6: Bei größeren Software-Projekten müssen oft Alternativen ausprobiert und ältere Software-Stände schnell wieder verfügbar werden. Dies wird durch Versionsmanagementsysteme unterstützt. Arbeiten Sie sich dazu in Git (<https://git-scm.com/>) ein, was auch die Arbeit in Teams an Praktikumsaufgaben wesentlich erleichtern kann. Ein Blick auf Github <https://github.com/> ist ebenfalls sinnvoll.

Aufgabe 7: Software läuft oft verteilt auf mehreren Rechnern. Um dies auf nur einem Rechner auszuprobieren kann eine Software wie Docker (<https://www.docker.com/>) sehr hilfreich sein. Erstellen Sie einen Docker-Container, installieren Sie Java und bringen Sie eine Ihrer Praktikumsaufgaben, die zur Ein- und Ausgabe nur die Konsole nutzt, zum Laufen.