

Übung 10: Design Patterns



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Software Engineering
WS 2018/19 - Dr. Michael Eichberg

Abgabe

Die Übung wird als *sbt*-Projekt (<http://www.scala-sbt.org>) bereitgestellt. *sbt* kann verwendet werden, um Java (und Scala) Projekte einfach auszuführen und zu testen. Installieren Sie *sbt* auf Ihrem Rechner und stellen Sie sicher, dass ein Java SDK (min. Java 8) installiert ist. Überprüfen Sie, dass der Java Compiler *javac* auf der Kommandozeile ausführbar ist, auch wenn Sie sich nicht im Verzeichnis mit der ausführbaren Datei befinden. Wenn nicht, passen Sie Ihren *PATH* entsprechend an, Hinweise wie dies bei Ihrem Betriebssystem möglich ist, finden Sie im Internet.

IntelliJ erlaubt das Importieren von *sbt*-Projekten direkt, dafür muss allerdings das Scala Plugin installiert sein. Um ein Eclipse-Projekt zu erzeugen, kann *sbt eclipse* im Projektverzeichnis (das Verzeichnis, das die Datei *build.sbt* enthält) ausgeführt werden, danach kann das Projekt mit Datei > Importieren > Vorhandene Projekte in Arbeitsbereich importiert werden. Wenn Sie eine *main*-Methode geschrieben haben, können Sie Ihr Programm mit *sbt run* ausführen, Tests können Sie unter *src/test/java/* anlegen und mit *sbt test* ausführen.

Das Ausführen des Kommandos *sbt* im Projektverzeichnis startet den interaktiven Modus von *sbt*. Im interaktiven Modus können Kommandos ausgeführt werden, ohne jedes Mal *sbt* eingeben zu müssen.

Um Ihre Lösung abzugeben, melden Sie sich zunächst unter

<https://submission.st.informatik.tu-darmstadt.de/course/se18>

an und erzeugen Sie ein *submission token*. Wenn Sie den interaktiven Modus von *sbt* verwenden, führen Sie dann folgendes Kommando aus:

```
submit <ihreTUID> <submissionToken>
```

wobei *<ihreTUID>* Ihre eindeutige Identifikationsnummer an der TU Darmstadt (**nicht Ihre Matrikelnummer!**) und *<submissionToken>* das zuvor generierte Token ist. Geben Sie die spitzen Klammern nicht mit an, der Befehl sollte etwa so aussehen: *submit ab01cdef 01234567*. Wenn Sie nicht den interaktiven Modus verwenden, muss das Kommando in Anführungszeichen gesetzt werden, also *sbt "submit <ihreTUID> <submissionToken>"*. Sie können (innerhalb der Abgabefrist) beliebig oft eine Lösung einreichen, allerdings wird nur die zuletzt eingereichte Lösung bewertet. Die letzte Lösung, die ein Gruppenmitglied eingereicht hat, wird zur Bewertung für die ganze Gruppe herangezogen. Koordinieren Sie sich daher in Ihrer Gruppe, wer Ihre gemeinsame Lösung einreicht.

Stellen Sie sicher, dass Sie das zur Verfügung gestellte Template nutzen und die Namen und Signaturen der vorgegeben Klassen und Methoden nicht verändern sowie dass von Ihnen hinzugefügte Klassen und Methoden die geforderten Namen und Signaturen verwenden. Ändern Sie außerdem nichts an der vorgegebenen Datei *build.sbt*. Andernfalls wird das System Ihre Lösung nicht bewerten können. Beachten Sie, dass der Zugriff auf die Abgabepattform nur im **internen Netz der Universität** möglich ist. Für einen Zugriff von außerhalb benötigen Sie daher eine VPN-Verbindung. Überprüfen Sie, ob Ihre Abgabe erfolgreich war, indem Sie sie von der Abgabepattform herunterladen. Überprüfen Sie dabei auch, ob alle Dateien in der Abgabe enthalten sind.

Einführung

In dieser Übung befassen Sie sich mit Design Patterns. Bearbeiten Sie die folgenden beiden Aufgaben und erstellen Sie eine einzelne PDF-Datei mit Ihren Antworten. Legen Sie diese Datei in den Ordner *solution* in dem Template, das Ihnen zur Verfügung gestellt wurde, und verwenden Sie zur Abgabe wie oben angegeben *sbt submit*.

Problem 1 Design Pattern erkennen**10P**

- a) Im zur Verfügung gestellten Template finden Sie eine Programm zur Sortierung von String-Arrays. Identifizieren Sie in diesem Template verwendete, Ihnen bekannte Design Pattern und nutzen Sie die Annotationen aus dem Package `ex10.annotations` um die einzelnen Komponenten der identifizierten Patterns (Klassen und Methoden) zu annotieren.
- b) Erstellen Sie ein Sequenzdiagramm, das ausgehend von der Methode `sort` der Klasse `Main` die Methodenaufrufe beim Sortieren eines Arrays darstellt.

Problem 2 Strategy Pattern implementieren**6P**

- a) Erweitern Sie die das vorgegebene Programm um die Funktionalität, Strings in den sortierten Arrays zu suchen. Verwenden Sie das Strategy Pattern, um den Suchalgorithmus variieren zu können. Implementieren Sie lineare und binäre Suche, wobei die Suchmethode ein String-Array und einen String entgegennehmen und den Index des gesuchten Strings zurückgeben soll (ist der String nicht im Array zu finden, soll -1 zurückgegeben werden). Erstellen Sie die neue(n) Klasse(n) im bereits existierenden Package `ex10.search`
- b) Annotieren Sie wie in Problem 1 Klassen und Methoden mit den passenden Annotationen für das Strategy Pattern.