

Datenbanken II

Wintersemester 2017/18

Programmieraufgabe 0

abzugeben bis: Montag, 30. Oktober 2017, 23:59 Uhr

1. Allgemeines

Geben Sie bitte Ihr fertiges Programm bis spätestens **Montag, 30. Oktober 2017, 23:59 Uhr**, über das Abgabesystem ¹ ab. Das System führt anschließend automatisierte Tests durch und Sie erhalten eine entsprechende Rückmeldung vom Abgabesystem.

Bitte beachten Sie, dass dies nur eine Test-Programmieraufgabe ist, sie fließt also nicht in die Benotung ein.

1.1. Betreuung

Bei Unklarheiten oder Verständnisproblemen gibt es für die Studierenden folgende Möglichkeiten diese abzuklären:

1. Slack Channel **#db2** ² (bevorzugter Kommunikationskanal)
2. Tutorium (Rene Thaler, montags, 16:00 - 17:00 im Rechnerübungsraum)
3. Sprechstunde: Dienstag, 09:00 - 10:00 Uhr, Büro 0.26 (Erdgeschoss)

Nehmen Sie *rechtzeitig* eine der Möglichkeiten wahr um Fragen abzuklären. Je früher Sie sich mit der Programmieraufgabe auseinandersetzen, desto früher können Sie Fragen stellen und desto einfacher ist es für die LV-Leitung bzw. den Tutor diese zu beantworten.

¹<https://abgaben.cosy.sbg.ac.at>

²<https://dbteaching.slack.com>

2. Aufgabenstellung

Um Sie mit unserem Abgabesystem ³ vertraut zu machen, werden Sie ein Echo-Programm implementieren. Dieses Echo-Programm soll überprüfen, ob mindestens ein Kommandozeilenparameter übergeben wurden. Ist dies der Fall, so wird der erste Kommandozeilenparameter auf `stdout` geschrieben.

Konkret sollen Sie die `main`-Methode in der Klasse `Main` implementieren (siehe `src/Main.java`).

Beispiele für das Verhalten des Echo-Programmes finden Sie in Anhang A dieses Dokumentes.

3. Codegerüst

Die Implementierung erfolgt in Java 8. Um Ihnen das Lösen der Programmieraufgabe zu erleichtern (und für automatisiertes Testen), stellen wir Ihnen ein Codegerüst zur Verfügung.

In dieser Test-Programmiernaufgabe, besteht das Codegerüst lediglich aus 1 Klasse, wobei Sie Ihren Code in der `main`-Methode der Datei `src/Main.java` einfügen müssen. Dies ist auch die einzige Datei, die Sie abgeben müssen.

Hinweis

Auch wenn Sie andere Dateien hochladen, werden diese nicht für die automatisierten Tests berücksichtigt.

3.1. Kompilieren & Ausführen

Sie können den Code wie folgt per Kommandozeile kompilieren und ausführen, wenn Sie sich im Order `src/` befinden:

```
javac Main.java
java Main <arg1> <arg2> ... <argN>
```

3.2. Ausgabeformat

Die Ausgabe erfolgt auf `stdout` und es wird lediglich das **erste** Kommandozeilenargument ausgegeben.

³<https://abgaben.cosy.sbg.ac.at>

4. Abgabe

Die Abgabe Ihrer Implementierung erfolgt über unser Abgabesystem ⁴. Beachten Sie, dass nur die jeweils **letzte** Abgabe gewertet wird.

Hinweis

Verwenden Sie in Ihrer Implementierung keinesfalls Sonderzeichen oder Umlaute. Dies kann zu unvorhergesehenen Fehlern führen.

4.1. Automatisierte Tests

Die Überprüfung Ihrer Implementierung erfolgt automatisiert. Hierbei wird zuerst überprüft, ob Ihr Programmcode kompiliert. Danach werden Tests mit verschiedenen Inputdaten durchgeführt. Der Output Ihres Programmes wird dann validiert. Daher weisen wir darauf hin, dass der Output genau so dargestellt werden muss, wie vorgegeben. Natürlich können Sie temporär zusätzlichen Output hinzufügen, allerdings muss dieser vor der Abgabe entfernt werden.

Sie finden den jeweiligen Beispiel-Output auch in Anhang A.

Nachdem die Tests abgeschlossen sind, bekommen Sie eine Erfolgs- bzw. Fehlermeldung über das Abgabesystem ⁴. Um Serverüberlastungen vorzubeugen, empfehlen wir, die Lösung nicht erst kurz vor der Abgabe-Deadline hochzuladen, sondern rechtzeitig abzugeben.

Während der Testphase wird Ihr Programm mit folgendem Kommandozeilenbefehl aufgerufen:

```
java Main ARG > outputfile
```

5. Zusätzliche Informationen

- *Getting Started* aus der Java Dokumentation (siehe ⁵)
- *Command-Line Arguments* aus der Java Dokumentation (siehe ⁶)

⁴<https://abgaben.cosy.sbg.ac.at>

⁵<http://docs.oracle.com/javase/tutorial/getStarted/index.html>

⁶<https://docs.oracle.com/javase/tutorial/essential/environment/cmdLineArgs.html>

Anhang A Beispiele

Im Folgenden finden Sie Beispiele für den Aufruf des Echo-Programmes mit dazugehöriger Ausgabe (auf `stdout`).

```
java Main Hallo  
Hallo
```

Abbildung 1: 1 Kommandozeilenargument

```
java Main Werte Studierende  
Werte
```

Abbildung 2: Mehr als 1 Kommandozeilenargument

```
java Main
```

Abbildung 3: Kein Kommandozeilenargument