

Programmieren und Software-Engineering I **Übung 7**

Name: _____ **Klasse:** _____ **Datum:** _____

Lernziele:

- Benutzereingabe
- Testdokumentation
- Teiler

Aufgabe 1: Tabelle (Ex_07_01_Table)

Erstelle ein Java-Programm, welches für die Zahlen 1,2, 3, ...100 jeweils die zweite und dritte Potenz in tabellarischer Form ausgibt.

<u>Beispiel:</u>	Zahl	Quadrat	Kubik
	1	1	1
	2	4	8
	3	9	27
	...		
	100	10000	1000000

Erweiterung:

- Für einen beliebigen Zahlenbereich
- Für beliebige Potenzen

Aufgabe 2: Echte Teiler (Ex_07_02_Divider)

Erstelle ein Ablauf- und ein Java-Programm, welches die Anzahl der **echten** Teiler (all die Zahlen, durch die man sie teilen kann, ohne dass ein Rest bleibt, außer der Zahl selbst) einer Zahl in folgender Form ausgibt. Die Zahl wird dabei vom Benutzer eingegeben!!

Beispiel: Die Zahl 41 hat 1 echte Teiler. // ohne Unterscheidung Einzahl-Mehrzahl
 ...
 Die Zahl 20 hat 5 echte Teiler.

<u>Code zum Einlesen:</u>	import java.util.*;	// nach dem Header und vor public class ...
	Scanner sc = new Scanner(System.in);	// einmal zu Beginn des Programms
	...	
	System.out.print("Zahl eingeben: ");	
	int number = sc.nextInt();	// sc kann öfter für Benutzereingabe // verwendet werden

Testdokumentation

Testdokumentation
Beschreibung wie das Programm getestet wurde.

Testfall	Zahl	Ausgabe erwartet	Ausgabe tatsächlich

Aufgabe 3: Teiler (EX 07 03 DividerII)

Erstelle ein Ablauf- und ein Java-Programm, welches für Zahlen in einem Bereich die Anzahl der Teiler in folgender Form ausgibt.

Beispiel: Die Zahl 5 hat 2 Teiler
Die Zahl 6 hat 4 Teiler
...
Die Zahl 41 hat 2 Teiler

Programmieren und Software-Engineering I

Übung 7

Aufgabe 4: Spiegelzahl (EX_07_04_MirrorNum)

Erstelle ein Java-Programm, welches alle dreiziffrigen Zahlen ausgibt, die von vorne und hinten gelesen gleich sind. Dabei werden immer genau 8 Zahlen in entsprechender Form pro Zeile ausgegeben.

Beispiel:

101	111	121	131	141	151	161	171
181	191	202	212				
...							

Erweiterung:

Die Ausgabe erfolgt für alle Spiegelzahlen in einem beliebigen Bereich. Zusätzlich soll die Anzahl der Zahlen pro Zeile variabel sein.

Aufgabe 5: Sternreihe (Ex_05_05_StarRow)

Erstelle ein Java-Programm, welches eine beliebig lange Sternreihe auf dem Bildschirm ausgibt. Dabei soll nach jedem fünften Stern ein Leerzeichen stehen.

Beispiel:

```
spaceAfterStars = 5;
numOfStars = 7;
Ausgabe: ***** **

numOfStars = 13;
Ausgabe: ***** ***** ***
```

Testdokumentation

Beschreibung wie das Programm getestet wurde.

Testfall	Leerz/Sterne	Ausgabe erwartet	Ausgabe tatsächlich

WICHTIG

Alle Programme müssen einen Programmkopf (=Beschreibung) enthalten.

z.B.:

```
/*
 * Name: Max Mustermann
 * Hü: 4
 * Bsp: 2
 * Datum: 20.10.2015
 * Dateiname: HUE_04_02_Dreieck.java
 * Beschreibung: Es wird für gegebene Seitenlängen a, b und c geprüft, ob es ein gleichseitiges, ein
 * gleichschenkeliges, ein rechtwinkeliges, ein sonstiges gültiges oder ein ungültiges
 * Dreieck ist.
 */
```

```
public class HUE_04_02_Dreieck { ...
```