

# Einführung in Java

## Übungen zur Wiederholung



### Zahlen

Schreibe ein Programm, welches die Eingabe dreier Zahlen erwartet und den Durchschnitt ausgibt.

### Verzweigungen

Schreibe ein Programm, das nach Abfrage des Wochentages (1=Montag, 7=Sonntag) je nach vorliegendem Fall „Schön :), Wochenende!“ bzw. „Schade, kein Wochenende :(“, ausgibt. Erstelle auch ein zugehöriges Struktogramm.

### Wiederholungen

Implementiere ein Programm, das ein Passwort solange abfragt, bis die (im Programm festgelegte) richtige Zeichenfolge eingegeben wird. Notiere ebenso ein passendes Struktogramm.

### Zeichenketten

Schreibe ein Programm, das ein beliebiges Wort einliest und es in umgekehrter (Buchstaben-) Reihenfolge wieder ausgibt.

### Array

Implementiere ein Programm, das eine Folge von 5 Zahlen in einen Array einliest und diesen im Anschluss kompakt ausgibt. Schließlich soll eine Zahl gesucht und – falls vorhanden – nach Bestätigung gelöscht werden dürfen.

### Klassen

Entwirf eine ganz einfache eigene Klasse mit grundlegenden Methoden und implementiere adäquate Ausgabe-Prozeduren oder wage Dich sogar an folgendes, (*deutlich*) komplexeres Projekt!

#### Projekt ToDo-Liste:

Eine **ToDo-Liste** soll mit dem Rechner verwaltet werden, weshalb eine entsprechende Klasse deklariert werden soll.

Eine **ToDo-Liste** soll dabei als erstes Attribut einen **Array** besitzen, der Elemente einer ebenfalls neu zu definierenden Klasse **Aufgabe** enthält, und als weiteres Attribut die Anzahl der aktuell aufgenommenen Aufgaben.

Eine **Aufgabe** soll aus einer **Beschreibung** sowie einer **Priorität** (niedrig, mittel, hoch) bestehen. Aufgaben mit höherer Priorität sollen bevorzugt abgearbeitet werden, Aufgaben mit gleicher Priorität der Reihenfolge ihres Eintrags entsprechend.

Die ToDo-Liste soll außerdem mittels einer Methode eine neue Aufgabe in die Liste aufnehmen können. Auch soll die erste Aufgabe angezeigt sowie (ggf. erst im Anschluss) gelöscht werden dürfen.

*(mögliche Erweiterungen: 1. eine weitere Liste, die die abgearbeiteten Aufgaben aufnimmt, 2. das Datum, an dem die Aufgabe bearbeitet wird, speichern, 3. Vererbung: Klasse Aufgabe spezialisieren)*