

## Programmieren I: Java

Wintersemester 2019/2020

### Lösungshinweise zu Programmierprüfung WS 2018/2019

(Zur selbstständigen Bearbeitung)

**Hinweise:** Es sind keine Hilfsmittel zugelassen. Nur leserliche und mit dokumentenechten Stiften geschriebene Antworten werden gewertet. Zum Bestehen müssen von den ersten drei Aufgaben min. zwei vollständig richtig beantwortet werden und von den letzten beiden Aufgaben mindestens eine.

Die Musterlösung ist lediglich als Lösungsvorschlag zu verstehen. Selbstverständlich können die einzelnen Aufgaben (in einem gewissen Rahmen) auch auf andere Arten gelöst werden. Grundsätzlich ist alles zulässig, was syntaktisch und semantisch korrekt ist und die Aufgabenstellung erfüllt!

#### Aufgabe 1

Schreiben Sie ein **lauffähiges Programm**, das in seiner **main**-Methode eine **boolesche** Variable deklariert und mit einem beliebigen Wert **initialisiert**. Der Wert der Variable soll anschließend auf der Konsole **ausgegeben** werden.

##### Lösungshinweise

```
public class A1{
    public static void main(String[] args) {
        boolean b = true;
        System.out.println(b);
    }
}
```

#### Aufgabe 2

Schreiben Sie eine **Klassen-Methode convert**, die eine Zahl vom Typ **double** entgegennimmt und als **Rückgabewert int** besitzt. Die Methode selbst soll den Wert aus dem Parameter in den Datentyp **int konvertieren** und das Ergebnis **zurückliefern** (nicht ausdrucken!).

##### Lösungshinweise

```
public static int convert(double d) {
    return (int) d;
}
```

### Aufgabe 3

Erzeugen Sie ein **zweidimensionales int-Feld** mithilfe der **statischen Erzeugung**, welches in der zweiten Dimension die Werte ( 4 ) bzw. ( 5 ; 6 ; 7 ) enthält.

#### Lösungshinweise

```
int[][] field = { { 4 }, { 5, 6, 7 } };
```

### Aufgabe 4

Schreiben Sie eine **Klasse** SuperFelder. Implementieren Sie eine **Instanz-Methode** druckeFeld, die alle Werte aus einem übergebenen **double-Feld** mit beliebiger Länge in folgender Form (Zahlenwerte sind nur Beispiele) auf der Konsole ausgibt: **(1.43 - 8.0 - 4.88 - 3.2)** inkl. Bindestriche, Klammern und ohne Zeilenumbruch.

Hinweis: Nach der letzten Ziffer in der Ausgabe soll kein weiterer Bindestrich stehen! Überlegen Sie sich zunächst, wie Sie dies umsetzen können.

#### Lösungshinweise

```
public class SuperFelder {
    public void druckeFeld(double[] field) {

        System.out.print( "(" + field[0] );

        for(int i = 1; i < field.length; i++) {
            System.out.print(" - " + field[i]);
        }

        System.out.println( ")" );
    }
}
```

## Aufgabe 5

Schreiben Sie eine **Klasse Schatz**, deren **Instanzen** jeweils über zwei von außen **nicht sichtbare Eigenschaften** **geheimnis** und **code** verfügen. Den Eigenschaften wird in einem **eigenen Konstruktor** bei der Objekterzeugung ein Wert zugewiesen. Schreiben Sie zusätzlich eine passende Methode **verraten**, die im Parameter einen Wert entgegen nimmt. Stimmt dieser Wert mit dem Wert von **code** überein, so soll das **geheimnis zurückgeliefert** werden. Andernfalls soll ein beliebiger Wert **zurückgeliefert** werden. Wählen Sie **passende Datentypen** für die Variablen aus.

### Lösungshinweise

```
public class Schatz {
    private String geheimnis;
    private int code;

    // Möglicher Konstruktor 1
    public Schatz() {
        geheimnis = "Irgendwas";
        code = 1234;
    }

    // Möglicher Konstruktor 2
    public Schatz (String geheimnis, int code) {
        this.geheimnis = geheimnis;
        this.code = code;
    }

    public String verraten(int code) {
        if(this.code == code)
            return geheimnis;
        else
            return "";
    }
}
```

Anmerkung: Einen eigenen Konstruktor zu definieren ist ausreichend für die Aufgabenstellung. Ob dieser Werte als Argumente entgegennimmt oder vordefinierte Werte verwendet, ist frei zu entscheiden.