

Bitte lösen Sie die Übungsaufgaben bis zum 28/29 Oktober. Wir besprechen sowohl das Aufgabenblatt 2 als auch Aufgabenblatt 3 in der nächsten Übung.

Aufgabe 3.1 (Datentypen) – Erstellen Sie eine Java-Klasse. Deklarieren und initialisieren Sie verschiedene Variablen mit einem gültigen Startwert! Geben Sie den jeweils aktuellen Wert auf der Konsole aus! Was versteht man unter der Bezeichnung „Camel Case“?

- Alter
- Name
- Einkommen
- Geschlecht (w/m/d)
- Vegetarier (ja/nein)
- Anzahl Menschen, die in Deutschland leben

Aufgabe 3.2 (Datentypen/Character) – Was ist die jeweilige Ausgabe? Versuchen Sie es zuerst „auf Papier“. Zur Überprüfung lassen Sie das Programm laufen.

```
public class CharacterBeispiele {  
  
    public static void main(String[] args) {  
  
        char zahl = 77;  
        char zeichen = 'M';  
  
        System.out.println(zahl);           // Ausgabe 1  
        System.out.println(zeichen);       // Ausgabe 2  
  
        System.out.println(zahl+1);        // Ausgabe 3  
        System.out.println(zeichen+1);     // Ausgabe 4  
  
        zahl = (char) (zahl+1);  
        System.out.println(zahl);          // Ausgabe 5  
        zeichen = (char) (zeichen + 1);  
        System.out.println(zeichen);       // Ausgabe 6  
  
    }  
}
```

Aufgabe 3.3 (Datentypen/Character) – Schreiben Sie ein Java-Programm, welches das Alphabet von A – Z ausgibt.

Aufgabe 3.4 (Datentypen/Casting) – Worin besteht der Unterschied zwischen Variante I und Variante II?

```
// Variante I
int testInt = 31 / 7;
float testFloat = 31 / 7;
double testDouble = 31 / 7;

System.out.println("testInt" + " " + testInt);    // Ausgabe 1
System.out.println("testInt" + " " + testFloat);  // Ausgabe 2
System.out.println("testInt" + " " + testDouble); // Ausgabe 3

// Variante II
int testInt = 31 / 3;
float testFloat = 31f / 3f;
double testDouble = 31d / 3d;

System.out.println("testInt" + " " + testInt);    // Ausgabe 4
System.out.println("testInt" + " " + testFloat);  // Ausgabe 5
System.out.println("testInt" + " " + testDouble); // Ausgabe 6
```

Aufgabe 3.5 (Dekrement/Inkrement-Operatoren) – Gegeben sei das folgende Java-Programm. Vollziehen Sie das Programm nach und überlegen Sie, welche Werte jeweils ausgegeben werden. Zur Überprüfung lassen Sie das Programm laufen.

```
public class InkrementDekrementUebung {  
    public static void main(String[] args) {  
        int x = 1, y = 2, z = 3;  
  
        System.out.println(x--);           // (1)  
        System.out.println(++x - x);       // (2)  
        System.out.println(x);             // (3)  
  
        if (x++ == 0) {  
            System.out.println (x);         // (4)  
        }  
        System.out.println(x);             // (5)  
  
        if ((y == 2) || (x++ > 0)) {  
            System.out.println (x);         // (6)  
        }  
        System.out.println(x);             // (7)  
        x += y;  
        System.out.println(x);             // (8)  
  
        if ((y == 2) | (x++ > 0)) {  
            System.out.println (x);         // (9)  
            z -= y;  
        }  
        System.out.println(x);             // (10)  
        System.out.println(y);             // (11)  
        System.out.println(z);             // (12)  
    } // main  
} // class
```

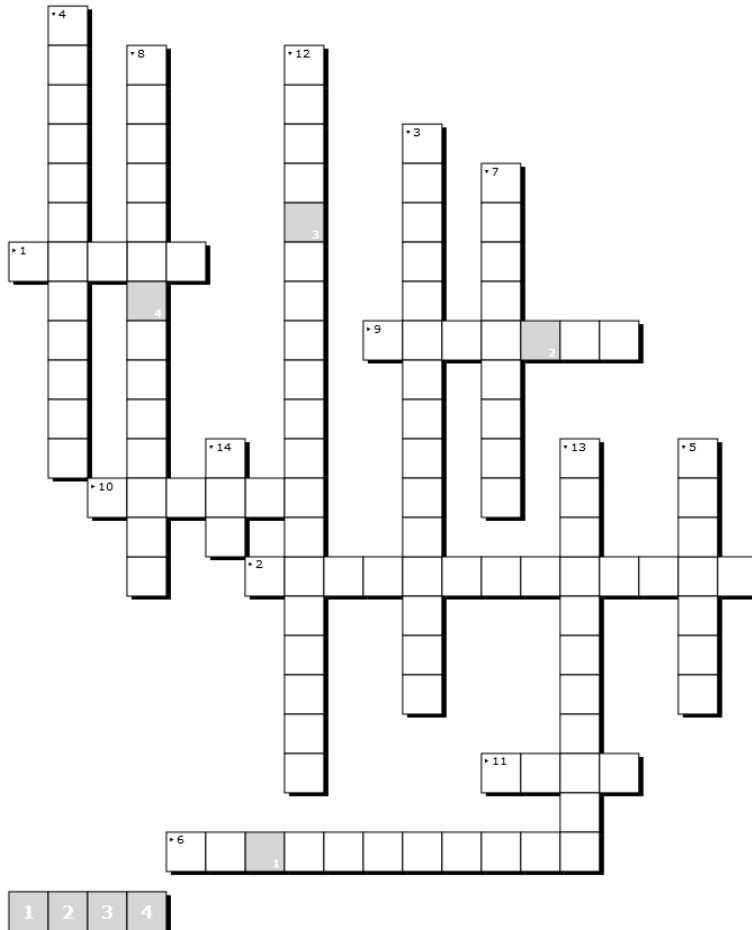
Aufgabe 3.6 (Kurzschluss-Operatoren) – Gegeben sei das folgende Java-Programm. Vollziehen Sie das Programm nach und überlegen Sie, was ausgegeben wird. Zur Überprüfung lassen Sie das Programm laufen.

```
public class KurzSchlussOperator {  
    public static void main(String[] args) {  
        char a = 'A';  
        char b = 'B';  
  
        if (a == --b) {  
            System.out.println ("1");  
        }  
  
        if ((a == b) || (a++ > b)) {  
            System.out.println("2");  
        }  
  
        System.out.println(a);  
    }  
}
```

PROGRAMMIER- TECHNIK I ≧ ÜBUNG ≦

3

Aufgabe 3.7 Kreuzworträtsel



1. Gibt es bei Compilern
2. Speichern komplexe Informationen
3. Erstmalige Wertzuweisung
4. Wird durch einen Datentyp definiert
5. Variablentyp für Zahlen
6. Damit gebe ich dem Compiler bekannt, dass es mich gibt
7. Mich kann man nicht mehr ändern
8. Von einer Programmiersprache reserviertes Wort
9. Typumwandlung
10. Gibt den Rest einer Division zurück
11. Mich muss man nur ein mal haben
12. Für eine Abkürzung gut
13. Ein Konzept in der Informatik
14. Fehler