

Übungsaufgaben Extra

Sequenzen / Auswahl / Mehrfachauswahl

Aufgaben zu Sequenzen

Aufgabe 1

Erstellen Sie ein ausführbares Programm mit dem Namen `Rechnen.java`. Das Programm führt Addition (+), Subtraktion (-), Multiplikation (*) und Division (/) mit den Zahlen 6 und 7 ohne den Einsatz von Variablen aus.

Die Bildschirmausgabe soll nachfolgendes Aussehen haben.

```
6 + 7 = 13
6 * 7 = 42
6 - 7 = -1
6 / 7 = 0
6.0 / 7.0 = 0.8571428571428571
6.0 / 7    = 0.8571428571428571
6 / 7.0    = 0.8571428571428571
```

Aufgabe 2

Erstellen Sie ein ausführbares Programm mit dem Namen `Modulo.java`. Das Programm führt mit dem Operator / (geteilt) und dem Modulo Operator % zwei Berechnungen aus und erzeugt nachfolgende

Bildschirmausgabe: `7 : 2 = 3 Rest 1`

Aufgabe 3

Erstellen Sie ein ausführbares Programm mit dem Namen `Modulo10Eigen.java`. Das Programm erzeugt untenstehende Bildschirmausgabe.

```
1023453 % 10 = 3
```

Allerdings dürfen Sie in Ihrem Programm nicht mit dem Modulo-Operator rechnen: `1023453 % 10 = 3`

Aufgabe 4

Erstelle ein ausführbares Programm mit dem Namen `Datentypen.java`. Das Programm weist den 8 primitiven Datentypen jeweils einen Wert zu und erzeugt nachfolgende

Bildschirmausgabe:

```
Wert für boolean: true
Wert für boolean: false
char: H
höchster Wert für byte: 127
kleinster Wert für byte: -128
höchster Wert für short: 32767
kleinster Wert für short: -32768
höchster Wert für int: 2147483647
kleinster Wert für int: -32768
höchster Wert für long: 9223372036854775807
höchster Wert für long: -9223372036854775808
float = 7.123457
double= 7.123456789012346
```

Aufgabe 5

Erstelle ein ausführbares Programm mit dem Namen `DreiecksTausch.java`. Gegeben ist die Quelltextzeile: `int a=7,`

b=13; Mit Hilfe einer Hilfsvariablen sollen beide Werte miteinander vertauscht werden. Erzeugen Sie danach nachfolgende Bildschirmausgabe: **a = 7 b = 13** **a = 13 b = 7**

Aufgabe 6

Erstellen Sie ein ausführbares Programm mit dem Namen KomaZahlKuerzen.java. Das Programm soll eine Doublevariable mit dem Wert 2.3897654 erzeugen. Im Anschluss soll sie algorithmisch auf zwei Stellen nach dem Komma gekürzt werden. Die Bildschirmausgabe der Zahl sollte wie nachfolgend dargestellt auf zwei Stellen nach dem Komma gekürzt sein: 2.38

Aufgabe 7

Erstellen Sie ein ausführbares Programm mit dem Namen Konstante.java. Das Programm legt eine Konstante mit dem Datentyp Double und dem Namen PI mit dem Wert 3.1415926535897932384626433832795 an. Danach soll diese Zahl auf dem Bildschirm ausgegeben werden. Berechnen Sie danach den Flächeninhalt eines Kreises mit dem Radius 5 Zentimeter und geben Sie das Ergebnis wie nachfolgend dargestellt auf dem Bildschirm aus:

PI lautet: 3.141592653589793
Der Kreis hat eine Fläche von 78.53981633974483
Quadratzentimetern.

Aufgabe 8

Erstellen Sie ein ausführbares Programm mit dem Namen Volumen.java. Das Programm erzeugt unten stehende Bildschirmausgabe:

Die Eingaben (10,9 und 3,1 und 3,5) sind in der Ausgabe kursiv und fett dargestellt.

Länge: 10,9
Breite: 3,1
Höhe: 3,5
Das Volumen beträgt: 118.265

Hinweis: Das Volumen berechnet sich mit der Formel:
Volumen = Länge * Breite * Höhe;

Aufgabe 9

Erstellen Sie ein ausführbares Programm mit dem Namen KreisGekuerzt.java, das die Eingabe des Radius und der Höhe verlangt. Danach soll die Fläche, der Umfang und das Volumen berechnet und auf dem Bildschirm wie nachfolgend dargestellt ausgegeben werden. Kürzen Sie das Ergebnis jeweils auf zwei Stellen nach dem Komma.

Die Eingaben (5 und 10) sind in der Ausgabe kursiv und fett dargestellt.

Bitte geben Sie den Radius ein: 5

Bitte geben Sie die Höhe ein: 10

PI = 3.14

Der Durchmesser beträgt 10.0

Die Fläche beträgt 78.53

Der Umfang beträgt 31.41

Das Volumen beträgt 785.39

Die Fläche eines Kreises berechnet man mit der Formel:

Radius * Radius * PI

Den Umfang eines Kreise berechnet man mit der Formel:

Radius * 2 * PI

Das Volumen einer Säule berechnet man mit der Formel:

Radius * Radius * PI * Höhe

Mehrseitige Bedingungsanweisung (if else if)

Aufgabe 10

Erstellen Sie ein ausführbares Programm mit dem Namen MehrseitigeBedingung.java. Das Programm prüft, ob die eingegebene Integer Variable alter kleiner ist

als 7. In diesem Fall wird „Geschäftsunfähigkeit“ auf dem Bildschirm ausgegeben.

als 18. In diesem Fall wird „beschränkte Geschäftsfähigkeit.“ ausgegeben.

Andernfalls wird „unbeschränkte Geschäftsfähigkeit.“ Ausgegeben.

Hinweis: Mehrseitige Bedingungsanweisungen enthalten einen if-else-if-Teil.

Aufgabe 11

Erstellen Sie ein ausführbares Programm mit dem Namen Noten.java. Das Programm erzeugt unten stehende Bildschirmausgabe.

In Abhängigkeit der Eingabe (hier die 1 fett und kursiv formatiert) erscheint die Note ausgeschrieben. Bei Eingabe einer ungültigen Note außerhalb von 1 - 6 erscheint: keine gültige Note

Bitte geben Sie eine Note ein: 1

sehr gut

Hinweis: 1 = sehr gut 2 = gut 3 = befriedigend 4 = ausreichend 5 = mangelhaft 6 = unbefriedigend“.

Aufgabe 12

Erstellen Sie ein ausführbares Programm mit dem Namen `BedingungsVerknuepfung.java`. Das Programm soll eine Integervariable `wert` mit den Bedingungsoperatoren UND (`&&`) und ODER (`||`) auf Wertebereiche prüfen:

Das nachfolgende Beispiel gilt für den Quelltextteil:

wert=200;

Wenn die Variable **wert**

zwischen 100 inklusive UND 200 inklusive liegt soll

`wert = 200` ausgegeben werden.

größer oder gleich 150 ist ODER kleiner oder gleich 10 ist soll

`wert = 200` ausgegeben werden.

Ihr Programm soll für den Quelltextteil: `wert=200;`

nachfolgende Bildschirmausgabe erzeugen.

1. `wert = 200`

2. `wert = 200`

3. *Ende*

Überlegen Sie sich für welches Zahlenspektrum von `wert` led.

Ende ausgegeben wird.

Aufgabe 13

Erstellen Sie ein ausführbares Programm mit dem Namen `SechsUnterschiedlicheLottoZahlen.java`. Das Programm erzeugt 6 unterschiedliche Lottozahlen zwischen 1 und 49 und gibt diese auf dem Bildschirm aus.

22 16 45 29 3 37

Hinweis: Verwenden Sie die Funktion `Math.random()`, die eine Zufallszahl zwischen 0 und 0,999 erzeugt. Durch Multiplikation von 49 und Addition der Zahl 1 gecastet in einen Integerwert erhalten Sie eine zufällige Lottozahl.

Aufgabe 14

Erstellen Sie ein ausführbares Programm mit dem Namen `GroessteZahl.java`.

Das Programm verlangt die Eingabe von drei Zahlen und gibt die größte der eingegebenen Zahlen auf dem Bildschirm aus. Bei den fett und kursiv dargestellten Zahlen handelt es sich um eine Benutzereingabe.

1. Zahl: **15**

2. Zahl: **99**

3. Zahl: **24**

Zahl 2 = 99 ist die größte Zahl.

Aufgabe 15

Erstellen Sie ein ausführbares Programm mit dem Namen GeradeUngerade.java. Das Programm soll prüfen, ob eine eingegebene Zahl gerade oder ungerade ist. Das Programm erzeugt unten stehende Bildschirmausgabe. Bei der fett und kursiv dargestellten Zahl handelt es sich um die Benutzereingabe.

Bitte geben Sie eine Zahl ein: 353

Die Zahl ist ungerade.

Hinweis: Verwenden Sie u.a. den Modulo- Operator (%).

Aufgabe 16

Erstellen Sie ein ausführbares Programm mit dem Namen BedingungsOperator.java. In Ihrem Quelltext sind die beiden nachfolgenden Quelltextzeilen gegeben:

```
int x=15, y=12; boolean zugang=false;
```

Verwenden Sie die Bedingungsoperatoren, um

a.) das Minimum der beiden Integerzahlen auszugeben

b.) die Ausgabe „offen“ bzw. „geschlossen“ in Abhängigkeit der Booleanvariable zu erzeugen.

Die Bildschirmausgabe sieht für obige Werte wie folgt aus:

12 Geschlossen

Aufgabe 17

Erstellen Sie ein ausführbares Programm mit dem Namen KommaNoteIfElse.java. Das Programm verlangt die Eingabe einer Kommanote und entscheidet dann bzgl. der Rundungsregeln über die verbale Note. Z.B. ergibt die Note 2,5 die verbale Note: befriedigend. Ungültige Noten, also Noten außerhalb des Notenspektrums zwischen 1 und 6 haben die Bildschirmausgabe: keine gültige Note zur Folge. Das Programm erzeugt untenstehende Bildschirmausgabe. Bei der fett und kursiv dargestellten Zahl handelt es sich um eine Benutzereingabe.

Bitte geben Sie eine Kommanote ein (z.B. 3,76): 1,45

sehr gut

Hinweis: 1 = sehr gut 2 = gut 3 = befriedigend 4 = ausreichend
5 = mangelhaft 6 = unbefriedigend

Aufgabe 18

Erstellen Sie ein ausführbares Programm mit dem Namen Temperatur.java. Schreiben Sie ein ausführbares Programm, das die Eingabe einer Gradzahl für die Temperatur verlangt. Danach soll je nach Gradzahl eine Ausgabe auf dem Bildschirm erfolgen:

Gradzahl	Ausgabe
<0 Grad	Ziehen Sie sich sehr warm an, es hat lediglich ... Grad.
>=0 aber <10	Heute ist es ziemlich kalt. Es hat .. Grad.
>=10 aber <=17	Erwägen Sie ein T-Shirt anzuziehen. Es hat heute .. Grad.
>=17 aber <=25	Ziehen Sie sich heute nicht zu warm an, bei .. Grad.
>=25	Machen Sie heute frei und gehen Sie zum Baden.

Aufgabe 19

Erstellen Sie ein ausführbares Programm mit dem Namen Schaltjahr.java. Zunächst erscheint untenstehender Dialog. Bei der fett und kursiv dargestellten Zahl handelt es sich um eine Benutzereingabe.

Bitte geben Sie eine Jahreszahl ein: **2004**

Bei dem Jahr **2000** handelt es sich um ein Schaltjahr. Ihr Programm gibt danach auf dem Bildschirm aus, ob es sich bei dem eingegebenen Jahr um ein Schaltjahr handelt oder nicht. Bei Eingabe der Jahreszahl:

*Bei dem Jahr **2000** handelt es sich um ein Schaltjahr.*

*Bei dem Jahr **1900** handelt es sich um kein Schaltjahr.*

Hinweis: Für das Bestimmen eines Schaltjahres gibt es drei

Regeln: Regel 1: Alle 4 Jahre ist ein Schaltjahr (das bedeutet: es gibt einen 29. Februar)

Regel 2: Alle 100 Jahre tritt Regel 1 außer Kraft.

Regel 3: Regel 2 tritt alle 400 Jahre außer Kraft.

Bemerkung: Testen Sie Ihr Programm mit folgenden Jahreszahlen:

Das Jahr 0 ist ein Schaltjahr

Das Jahr 1000 ist kein Schaltjahr

Das Jahr 1200 ist ein Schaltjahr

Das Jahr 1900 ist kein Schaltjahr

Das Jahr 1993 ist kein Schaltjahr

Das Jahr 2000 ist ein Schaltjahr

Das Jahr 2004 ist ein Schaltjahr

Aufgabe 20

Erstellen Sie ein ausführbares Programm mit dem Namen ABCsortieren.java. Gegeben ist die Quelltextzeile:

```
int hilf, a=3, b=1, c=2;
```

Erstellen Sie einen Quelltext, der die Werte der Variablen so miteinander tauscht, dass die Werte danach sortiert sind. a enthält danach den kleinsten und c den größten Wert.

Ihre Bildschirmausgabe sieht so aus: vorher:

a=3 b=1 c=2 nachher: a=1 b=2 c=3

Überlegen Sie sich geeignete Testfälle um Ihren Sortieralgorithmus zu testen.

Hinweis: Mögliche TESTFÄLLE

a = 1; b = 2; c = 3;	a = 1; b = 3; c = 2;
a = 2; b = 1; c = 3;	a = 2; b = 3; c = 1;
a = 3; b = 1; c = 2;	a = 3; b = 2; c = 1;
a = 1; b = 1; c = 1;	a = 88; b = 4; c = 88;
a = -3; b = -10; c = 12;	a = 0; b = 0; c = 0;

Mehrseitige Bedingungsanweisung (switch case)

Aufgabe 21

- a) Erstellen Sie ein ausführbares Programm mit dem Namen SwitchCase.java. Das Programm erstellt zunächst eine Character-Variable mit dem Namen hex und dem Buchstabenwert 'C'. Danach wird mit einer Switch-Case-Anweisung der entsprechende Dezimale Zahlenwert auf dem Bildschirm ausgegeben. In diesem Fall erscheint:

Dezimal = 12

Sollte der Variablen hex ein ungültiger Hexadezimalwert außerhalb von A-F zugewiesen werden, erscheint die Bildschirmausgabe:

Sie haben keinen gültigen Wert eingegeben!!!

- b) Erweitern Sie die obige Aufgabe um die Werte 1 - 9 und lassen sich einen beliebigen Dezimalwert als Hexadezimalwert anzeigen.

Die Bildschirmausgabe soll jetzt so aussehen:

Eingegebener	Wert :	9	10	11	12	13	14	15
Ausgabe	Hex =	9	A	B	C	D	E	F

- c) Erweitern Sie obigen Quellcode jetzt um die Logik einer Umrechnung jeder beliebigen Dezimalzahl in eine Hexadezimale Zahl.

Beispiel 45791 dezimal in B2DF Hexadezimal

Hinweis : Dezimalzahlen werden in Hexadezimal umgerechnet in dem die Dezimalzahl durch 16 dividiert und der Rest als Hex geschrieben

```
45791 / 16 = 2861 R 15 = F
2861 / 16  = 178  R 13 = D
178 / 16   = 11   R 2  = 2
11 / 16    = 0    R 11 = B
```

Aufgabe 22

Erstellen Sie ein ausführbares Programm mit dem Namen Schokoladenautomat.java. Ihr Programm stellt über nachfolgendes Auswahlmenü folgende Schokoladen zur Auswahl:

- 1.) Zartbitterschokolade
- 2.) Vollmilkschokolade
- 3.) weiße Schokolade
- 4.) Traubenschokolade

Nach Eingabe einer Nummer wird mittels einer Switch-Case-Anweisung der Preis angezeigt. Für Position

- 1 = 1,50 Euro,
- 2 = 1,10 Euro ,
- 3 = 1,20 Euro ,
- 4 = 1,30 Euro.

Für ungültige Nummern erscheint die Bildschirmausgabe:

„Ungültige Eingabe!“.

Aufgabe 23

Erstellen Sie ein ausführbares Programm mit dem Namen MonatsTage.java. Gegeben ist die Quelltextzeile:

```
int monat=2, jahr=2015, tage=0;
```

Ihr Programm erzeugt unten stehende Bildschirmausgabe.

Der 2. Monat im Jahr 2015 hatte 28 Tage

Hinweis: Verwenden Sie in Ihrem Programm sowohl die Switch-Case-Anweisung als auch die if-else-Anweisung. Sie können diese Aufgabe auch mit der Schaltjahr Aufgabe vereinen.

Aufgabe 24

Erstellen Sie ein ausführbares Programm mit dem Namen DatumGueltig.java. Das Programm prüft, ob ein eingegebenes Datum gültig ist oder nicht. Das Programm erzeugt unten stehende Bildschirmausgabe. Bei den fett und kursiv dargestellten Zahlen handelt es sich um eine Benutzereingabe.

Tag eingeben: **31**

Monat eingeben: **1**

Jahr eingeben: **2016**

Das Datum: 31.1.2016 ist gültig

Hinweis: Verwenden Sie in Ihrem Programm sowohl die if-else-Anweisung als auch die Switch-Case-Anweisung.

Aufgabe 25

Erstellen Sie ein ausführbares Programm mit dem Namen Taschenrechner.java. Das Programm erzeugt unten stehende Bildschirmausgabe. Bei den fett und kursiv dargestellten Zahlen handelt es sich um eine Benutzereingabe:

Bitte geben Sie die 1. ganze Zahl ein: **97**

Bitte geben Sie die 2. ganze Zahl ein: **45**

1.) Addition

2.) Subtraktion

3.) Multiplikation

4.) Division

2

Ausgabe : 97.0 - 45.0 = 52.0