

Übung 2.2.7 #1

Schreibt eine Methode

`public void pruefeZahl(int zahl)`

die mit `System.out.println()` ausgibt, ob die eingegebene Zahl gerade oder ungerade ist. Verwendet dabei den Modulo-Operator `%`.

```
public void pruefeZahl(int zahl)
{
    if (zahl % 2 == 0)
        System.out.println("gerade");
    else
        System.out.println("ungerade");
}
```

Übung 2.2.7 #2

Schreibt eine Methode

`public void pruefeVorzeichen(int zahl)`

die ausgibt, ob die eingegebene Zahl positiv, negativ oder null ist.

```
public void pruefeVorzeichen(int zahl)
{
    if (zahl < 0)
        System.out.println("negativ");
    else if (zahl > 0)
        System.out.println("positiv");
    else
        System.out.println("null");
}
```

Übung 2.2.7 #3

Schreibt eine Methode

`public int getMinimum(int a, int b)`

die die kleinere Zahl mit return zurückliefert

```
public int getMinimum(int a, int b)
{
    if (a < b)
        return a;
    else
        return b;
}
```

Übung 2.2.7 #4

Schreibt eine Methode

`public void analysiereZahlen(int a, int b)`

die mit `System.out.println()` die größere Zahl ausgibt.

Falls beide Zahlen gleich sind, soll eine entsprechende Meldung ausgegeben werden.

```
public void analysiereZahlen(int a, int b)
{
    if (a > b)
        System.out.println("a > b");
    else if (b > a)
        System.out.println("b > a");
    else
        System.out.println("a = b");
}
```

Übung 2.2.7 #5

Schreibt eine Methode

`public void pruefeAlter(int alter)`

die ausgibt, ob eine Person

1. noch minderjährig ist (unter 18)
2. volljährig ist (18 oder älter)
3. oder bereits das Rentenalter erreicht hat (ab 67)

```
public void pruefeAlter(int alter)
{
    if (alter < 18)
        System.out.println("minderjährig");
    else if (alter < 67)
        System.out.println("volljährig");
    else
        System.out.println("Rentner");
}
```

Übung 2.2.7 #6

Schreibt eine Methode

`public void bewerteNote(int note)`

die zu einer Schulnote (1 bis 6) einen Text ausgibt, zum Beispiel "sehr gut" für die Note 1. Für ungültige Eingaben von note sollte eine Fehlermeldung ausgegeben werden.

```
public void bewerteNote(int note)
{
    if (note == 1)
        System.out.println("sehr gut");
    else if (note == 2)
        System.out.println("gut");
    else if (note == 3)
        System.out.println("befriedigend");
    else if (note == 4)
        System.out.println("ausreichend");
    else if (note == 5)
        System.out.println("mangelhaft");
    else if (note == 6)
        System.out.println("ungenügend");
    else
        System.out.println("Fehler");
}
```

Übung 2.2.7 #7

Schreibt eine Methode

`public int getMaximum(int a, int b, int c)`

die die größte der drei Zahlen zurückliefert

```
public int getMaximum(int a, int b, int c)
{
    int max = a;
    if (b > max) max = b;
    if (c > max) max = c;
    return max;
}
```

Übung 2.2.7 #7

Schreibt eine Methode

`public int getMaximum(int a, int b, int c)`

die die größte der drei Zahlen zurückliefert

Profi-Lösung



```
public int getMaximum(int a, int b, int c)
{
    return Math.max(a, Math.max(b, c));
}
```

Übung 2.2.7 #8

Schreibt eine Methode

```
public String  
    getTemperaturBewertung(double temp)
```

die einen String zurückliefert, der die übergebene Temperatur bewertet:

- unter 0 °C → "Gefrierpunkt unterschritten"
- 0 °C bis unter 10 °C → "sehr kalt"
- 10 °C bis unter 20 °C → "kühl"
- 20 °C bis unter 27 °C → "angenehm"
- 28 °C bis unter 35 °C → "heiß"
- 35 °C und mehr → "sehr heiß, Hitzegefahr!"

```
public String  
    getTemperaturBewertung(double temp)  
{  
    if (temp < 0)  
        return "Gefrierpunkt unterschritten";  
    if (temp < 10)  
        return "sehr kalt";  
    if (temp < 20)  
        return "kühl";  
    if (temp < 27)  
        return "angenehm";  
    if (temp < 35)  
        return "heiß";  
  
    return "sehr heiß, Hitzegefahr!";  
}
```

Übung 2.2.7 #9

Schreibt eine Methode

```
public boolean pruefeDatum  
    (int tag, int monat, int jahr)
```

die true zurückliefert, wenn das übergebene Datum korrekt ist. Auf Schaltjahre müsst ihr hier keine Rücksicht nehmen. April, Juni, September und November haben nur 30 Tage.

Recht umständliche Lösung



```
public boolean  
    pruefeDatum(int tag, int monat, int jahr)  
{  
    if (monat == 4 || monat == 6 ||  
        monat == 9 || monat == 11)  
    {  
        if (tag > 0 && tag <= 30)  
            return true;  
    }  
    else if (monat == 2)  
    {  
        if (tag > 0 && tag <= 28)  
            return true;  
    }  
    else  
    {  
        if (tag > 0 && tag <= 31)  
            return true;  
    }  
  
    return false;  
}
```

Übung 2.2.7 #9

Schreibt eine Methode

`public boolean pruefeDatum`

`(int tag, int monat, int jahr)`

die true zurückliefert, wenn das übergebene Datum korrekt ist. Auf Schaltjahre müsst ihr hier keine Rücksicht nehmen. April, Juni, September und November haben nur 30 Tage.

Recht bessere Lösung



```
public boolean
    pruefeDatum(int tag, int monat, int jahr)
{
    int maxDay;

    if (monat == 4 || monat == 6 ||
        monat == 9 || monat == 11)
        maxDay = 30;
    else if (monat == 2)
        maxDay = 28;
    else
        maxDay = 31;

    if (tag > 0 && tag <= maxDay)
        return true;
    else
        return false;
}
```

Übung 2.2.7 #9

Schreibt eine Methode

```
public boolean pruefeDatum
```

```
    (int tag, int monat, int jahr)
```

die true zurückliefert, wenn das übergebene Datum korrekt ist. Auf Schaltjahre müsst ihr hier keine Rücksicht nehmen. April, Juni, September und November haben nur 30 Tage.

Noch bessere Lösung



```
public boolean pruefeDatum
    (int tag, int monat, int jahr)
{
    int maxDay = 31;

    if (monat == 4 || monat == 6 ||
        monat == 9 || monat == 11)
        maxDay = 30;
    else if (monat == 2)
        maxDay = 28;

    return (tag > 0 && tag <= maxDay);
}
```

Übung 2.2.7 #10

Schreibt eine Methode

```
public int getMiniAus4
```

```
(int a, int b, int c, int d)
```

die das Minimum der vier Zahlen zurückliefert. Es gibt hier zwei Lösungsmöglichkeiten!

Gute Lösung



```
public int  
    getMiniAus4(int a, int b, int c, int d)  
{  
    int mini = a;  
    if (b < mini) mini = b;  
    if (c < mini) mini = c;  
    if (d < mini) mini = d;  
    return mini;  
}
```

Übung 2.2.7 #10

Schreibt eine Methode

```
public int getMiniAus4
```

```
(int a, int b, int c, int d)
```

die das Minimum der vier Zahlen zurückliefert. Es gibt hier zwei Lösungsmöglichkeiten!

Gute Lösung



```
public int  
    getMiniAus4(int a, int b, int c, int d)  
{  
    return  
        getMinimum(  
            getMinimum(a,b),  
            getMinimum(c,d)  
        );  
}
```