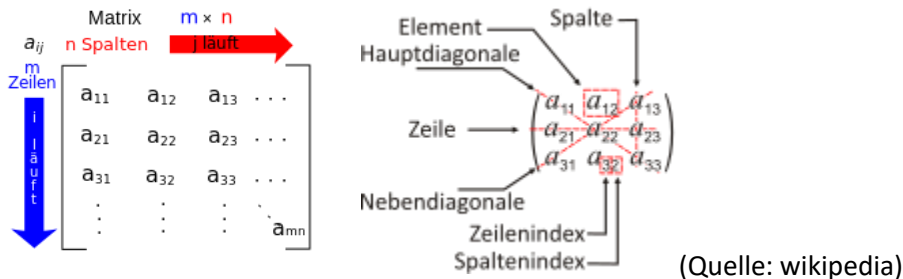


## Matrizen

In der Mathematik versteht man unter einer Matrix (Plural Matrizen) eine rechteckige Anordnung (Tabelle) von Elementen (meist mathematischer Objekte, etwa Zahlen). Mit diesen Objekten lässt sich dann in bestimmter Weise rechnen, indem man Matrizen z. B. addiert oder miteinander multipliziert.



## Mehrdimensionale Felder

Analog zu dem schon bekannten Datentyp Feld (bisher nur mit einer Dimension) lassen sich Felder mit (theoretisch) beliebig vielen Dimensionen implementieren. In Java werden mehrdimensionale Arrays durch Arrays von Arrays realisiert. Benötigen wir z. B. ein zweidimensionales Feld mit einer Breite von 15 und einer Länge von 20 Elementen muss ein Feld der Länge 15 vereinbart werden, welches jeweils Felder der Länge 20 enthält. Diese Art der Vereinbarung mehrdimensionaler Felder bietet u. a. auch die Möglichkeit nichtrechteckige Felder zu realisieren.

Für unser Problem benötigen wir ein Feld von 3 x 3 Integerzahlen.

```
int[][] matrix = new int[3][3];
```

Um später darauf zugreifen zu können sollten die Anzahl der Spalten und Zeilen in einer Variablen gespeichert werden. Hierbei ist zu beachten, dass zuerst die Zeile, dann die Spalte angegeben wird.

```
int zeilen = 3, spalten = 3;
int[][] matrix = new int[zeilen][spalten];
```

Später können die einzelnen Elemente mit Werten belegt werden:

```
matrix[0][0] = 4; matrix[0][1] = 10; matrix[0][2] = 3;
matrix[1][0] = 12; matrix[1][1] = 1; matrix[1][2] = 18;
matrix[2][0] = 3; matrix[2][1] = 13; matrix[2][2] = 9;
```

Felder können auch (aber NUR bei der Initialisierung) direkt gefüllt werden. Die Dimensionen des Feldes werden dann automatisch erkannt. Sollten die Werte zu Beginn bekannt sein, ist eine Belegung so möglich:

```
public static int[][] matrix = { { 4, 10, 3 },
                                  { 12, 1, 18 },
                                  { 3, 13, 9 } };
```

### Aufgabe 1

Analysieren Sie das gegebene Programm FeldMatrix. Beschreiben Sie die Aufgaben und die Arbeitsweise der einzelnen Methoden. Was genau gibt das Programm aus? Testen Sie danach die Funktionsweise.

```
public class FeldMatrix {

    public static int summe(int[][] m, int zeile) {
        int s = 0;
        for (int i = 0; i < m[zeile].length; i++) {
            s = s + m[zeile][i];
        }
        return s;
    }

    public static void ausgeben(int[][] m) {
        for (int y = 0; y < m.length; y++) {
            for (int x = 0; x < m[y].length; x++) {
                System.out.printf("%5d",m[y][x]);
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        int[][] matrix = { { 4,10, 3,20 },
                           {12, 1,18, 3 },
                           { 3,13, 9,10 },
                           {25, 9,12, 6 } };

        ausgeben(matrix);
        System.out.println();
        for (int i = 1; i <= matrix.length; i++) {
            System.out.println("Summe Zeile " + i + ": " + summe(matrix,i-1));
        }
    }
}
```

## Aufgabe 2

Implementieren Sie in das gegebene Programm aus Aufgabe 1:

- Die Spaltensummen sollen berechnet und ausgegeben werden.
- Die Summe der beiden Diagonalen soll berechnet werden.
- Informieren Sie sich über magische Quadrate und implementieren Sie eine Methode `public static boolean istMagisch(int[][] m)` die testet, ob `matrix` ein magisches Quadrat darstellt. Testen Sie mit verschiedenen großen magischen Quadraten.

## Aufgabe 3

Spiegelt man die Elemente des magischen Quadrats an der senkrechten Mittellinie, so erhält man wieder ein magisches Quadrat. Implementieren Sie eine entsprechende Methode

```
public static void spiegelnAnSenk(int[][] m){...}
```

die die übergebene Matrix entsprechend verändert.

Beispiele:  $\text{spiegeln\_an\_senk} \begin{pmatrix} 2 & 7 & 6 \\ 9 & 5 & 1 \\ 4 & 3 & 8 \end{pmatrix} = \begin{pmatrix} 6 & 7 & 2 \\ 1 & 5 & 9 \\ 8 & 3 & 4 \end{pmatrix},$

$$\text{spiegeln\_an\_senk} \begin{pmatrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{pmatrix} = \begin{pmatrix} 13 & 2 & 3 & 16 \\ 8 & 11 & 10 & 5 \\ 12 & 7 & 6 & 9 \\ 1 & 14 & 15 & 4 \end{pmatrix}$$

## Aufgabe 4 (Ergänzung)

In der Mathematik gibt es eine Menge von Operationen auf Matrizen.

Informieren Sie sich im Internet über die Regeln für das Berechnen des jeweiligen Ergebnisses und implementieren Sie ein geeignetes Java-Programm:

- Addition zweier Matrizen
- Multiplikation einer Matrix mit einer Zahl
- Skalarprodukt zweier Matrizen
- Vektorprodukt (Kreuzprodukt) zweier Matrizen
- Determinante einer Matrix