Array

Inhaltsverzeichnis

TECHNISCHE UNIVERSITÄT DARMSTADT

Das Array

TECHNISCHE UNIVERSITÄT DARMSTADT

- Variablen gleichzeitig und komfortabel manipulieren

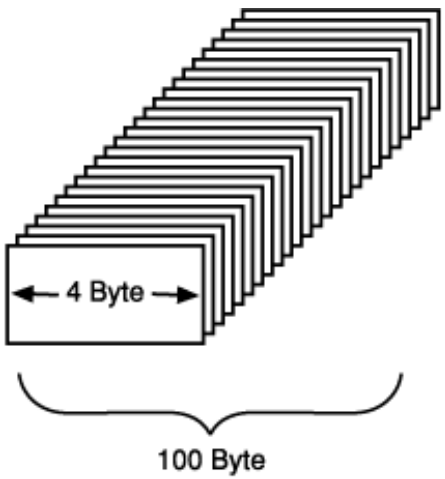
```
int var1 = 10;
int var2 = 6;
int var3 = 21;

var1++;
var2--;
var3++;
```

- Daher gibt es in Java den Datentyp Array

Deklarieren

TECHNISCHE UNIVERSITÄT DARMSTADT

- Was versteht man unter dem Begriff Array?
- Feld bzw. Container, das mehrere Objekte vom gleichen Typ aufnehmen kann
- Schuhkarton-Prinzip



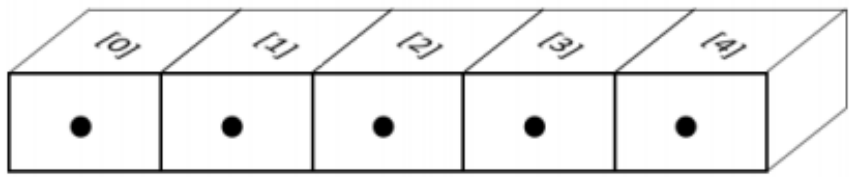

```
long[]  myArray = new long[25];
```

Abbildung: Array [1]

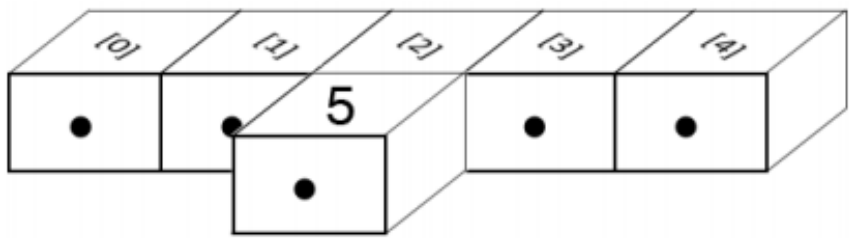
Deklarieren
Direkte Initialisierung

TECHNISCHE UNIVERSITÄT DARMSTADT

- `int[] myArray = {2,2,5,9,1092}`



- Mit myArray[2] erhählt man den Wert an Stelle **2**
  - `int value  = 3 + myArray[2];`



Index-Zugriff

TECHNISCHE UNIVERSITÄT DARMSTADT

- Array mit `int [] myArray = new int[10]` erstellt
- Elemente sind von 0 bis 9 durchnummeriert

Abbildung: Array mit 10 Elementen

- myArray[i] gibt also das **i+1**te Element zurück
- myArray.length gibt die Anzahl der Elemente zurück
  - **Achtung:** Vergleiche myString.length()

Achtung

Informatiker fangen immer bei 0 an zu zählen!

Index-Zugriff
Zugriffsfehler

TECHNISCHE UNIVERSITÄT DARMSTADT

```
int[] stats = {1,2,3,4};

for(int i = 0; i < 5; i++){
        System.out.println(stats[i]);
}
/*
Ausgabe:
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException
    : 4
        at Start.main(Start.java:12)1
2
3
4
*/
```

For vs. foreach Schleife

TECHNISCHE UNIVERSITÄT DARMSTADT

- Häufig möchte man nur durch ein Array iterieren

```
String[] data = { "Toronto", "Stockholm" };
for(int i = 0; i< 2;i++)
   System.out.println(data[i]);
```

- Deshalb foreach-Schleife

```
String[] data = { "Toronto", "Stockholm" };
for(String s : data)
   System.out.println(s);
```

Ein Array kopieren
Erster Versuch

TECHNISCHE UNIVERSITÄT DARMSTADT

```
public static void main(String[] args) {
    int[] myArray = new int[2];
    int[]  copy = myArray;

    myArray[0] = 0;
    myArray[1] = 1;

    System.out.println(myArray[0]); //0
    System.out.println(myArray[1]); //1

    copy[1] = 2;

    System.out.println(myArray[1]); //2
}
```

Ein Array kopieren
Analyse

TECHNISCHE UNIVERSITÄT DARMSTADT

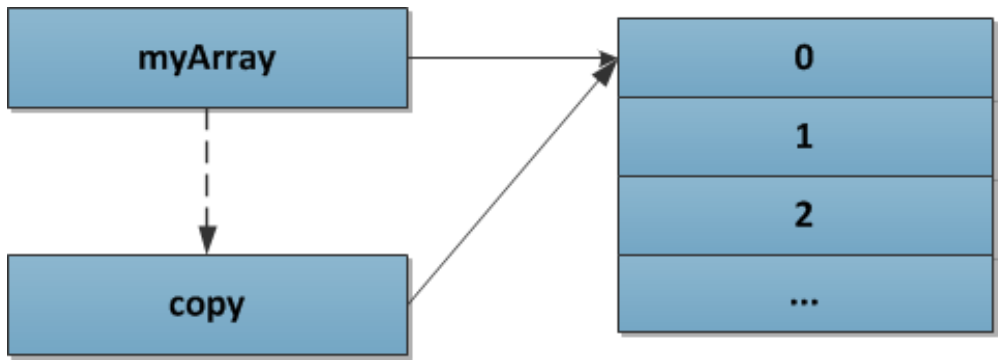


Abbildung: Referenzierung [2]

- Bei der Zuweisung mit **=** wurde nur die Zieladresse kopiert
- Deshalb wird das gleiche Array referenziert

Ein Array kopieren
Zweiter Versuch

TECHNISCHE UNIVERSITÄT DARMSTADT

```
public static void main(String[] args) {
    int[] myArray = {7,6,5,4,3};
    int[]  copy = {0,0,0,0,0};

    //arraycopy(source,startIndex,destination,destIndex,length)
    System.arraycopy(myArray, 1, copy, 1, 3);

    for(int i=0; i < copy.length; i++)
      System.out.println(copy[i]); //0,6,5,4,0
}
```

Ein Array kopieren
Analyse

TECHNISCHE UNIVERSITÄT DARMSTADT

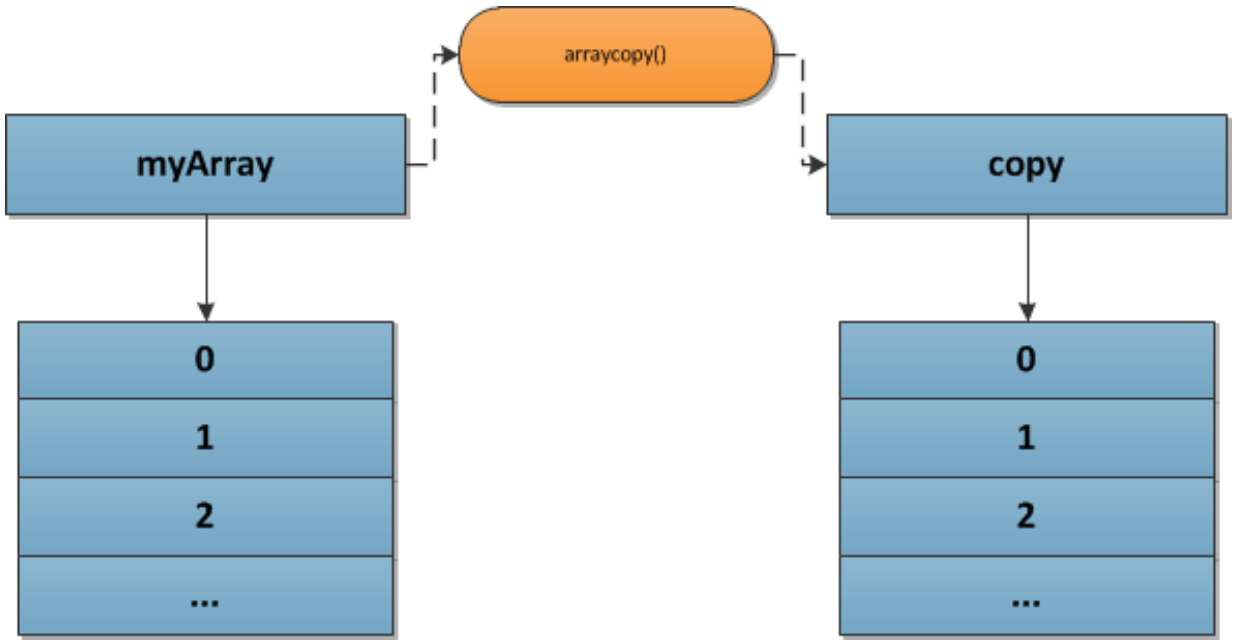
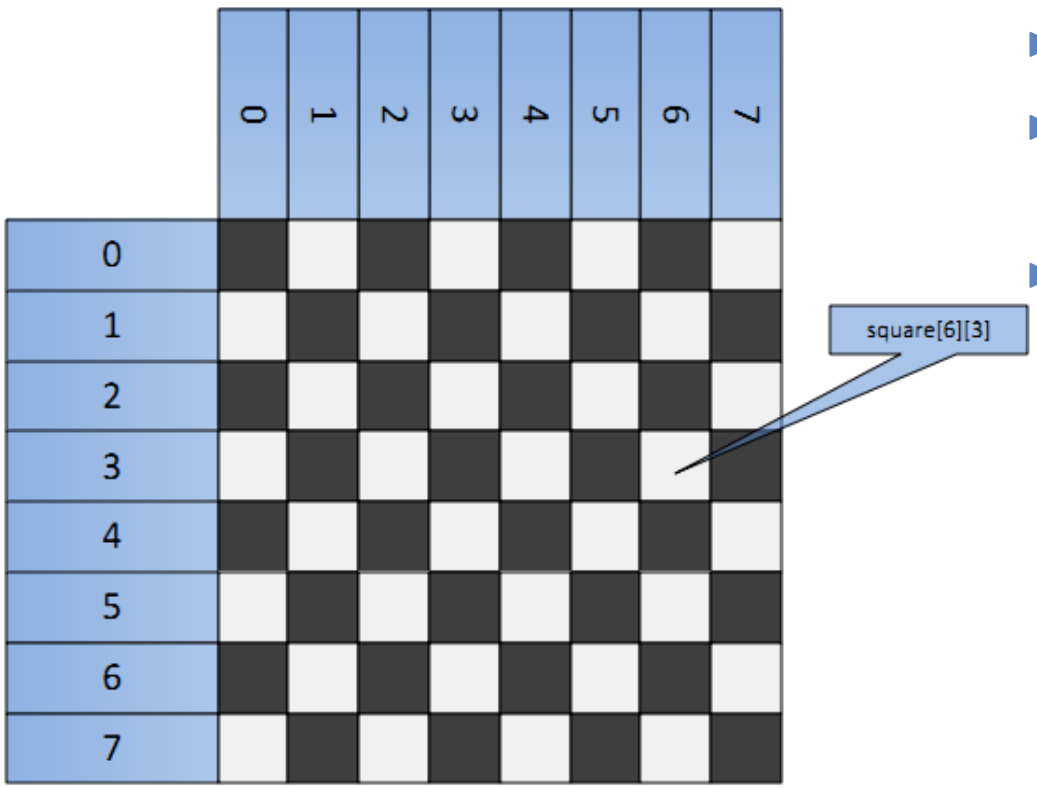


Abbildung: Referenzierung [2]

- Copy liegt nun in einem neuen unabhängigen Speicherbereich

Mehrdimensionale Arrays

TECHNISCHE UNIVERSITÄT DARMSTADT



- `int [][] square = new int [8][8];`
- Jede Dimension stellt man durch einen eigenen Index dar
- Adressierung: `square[Spalte][Zeile]`

Initialisieren

TECHNISCHE UNIVERSITÄT DARMSTADT

```
//1. Direkte Initialisierung
int[][] table = {{1,2}, {3,4}, {5,6}, {7,8}};
//2.
int[][] array = new int[2][2];
array[0][0] = 10;
array[0][1] = 20;
array[1][0] = 30;
array[1][1] = 40;
//3.
int[][] chessBoard = new int[8][8];
for(int y = 0;y < 8;y++){
    for(int x = 0;x < 8;x++){
        chessBoard[x][y] = 10;
    }
}
```

Fragerunde

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wie erhält man Zugriff auf die Länge eines **Arrays**/**Strings**?
- Wie wird ein zweidimensionales Array adressiert? (Stichwort: Zeile/Spalte)
- Wofür benötigt man arraycopy und warum erzielt eine einfache Zuweisung nicht den gewünschten Effekt?

Quellen

TECHNISCHE UNIVERSITÄT DARMSTADT

- **[1]** C++ in 21 Tagen
- **[2]** http://www.programmersbase.net