

# Software Development

Datenstrukturen

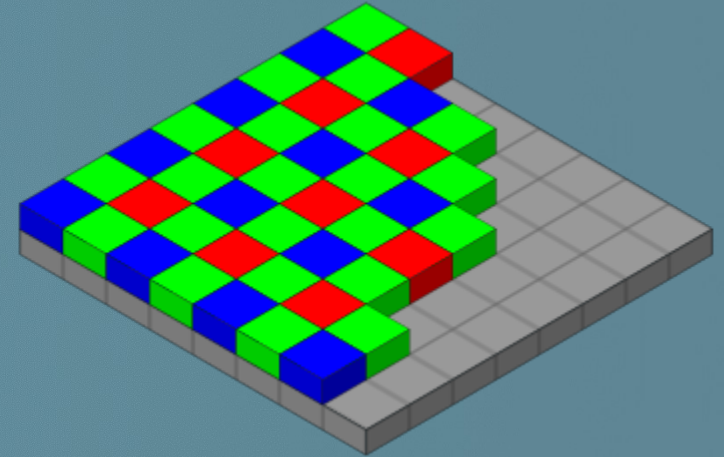
Software-Lösungen. Einfach. Clever.



# Inhalt

- Arrays
- Mehrdimensionale Arrays
- Listen für identische Datentypen (List)
- Listen für unterschiedliche Datentypen (ArrayList)

# Arrays



Software-Lösungen. Einfach. Clever.

# Arrays

- Speichern von mehreren Variablen des gleichen Typs in eine Datenstruktur
- Array kann eindimensional, mehrdimensional oder verzweigt sein
- Anzahl Dimensionen und Länge der Dimensionen bei der Instanziierung festgelegt. Werte können während der Lebensdauer der Instanz nicht geändert werden
- Arrays sind nullbasiert (Index beginnt bei 0 und endet bei n-1)
- Arrayelemente können beliebigen Typ aufweisen

# Arrays

## Deklaration eines einfachen Array

C #

```
int[] myArray = new int[5];  
myArray[1] = 5;  
Console.WriteLine (myArray[1]);
```

mit Werten

C #

```
int[] myArray = new int[] { 1, 3, 5, 7, 9 };  
Console.WriteLine (myArray[1]);
```

Alternative Schreibweise

C #

```
int[] myArray = { 1, 3, 5, 7, 9 };  
Console.WriteLine (myArray[0]);
```

0	1	2	3	4
	5			

0	1	2	3	5
1	3	5	7	9

# Arrays

## Deklaration eines mehrfachen Array

C #

```
int[,] myArray = new int[2, 3]; // 2 Zeile, 3 Spalten  
myArray[0, 1] = 5; // Index beginnt bei 0  
Console.WriteLine (myArray[0, 1]);
```

	0	1	2
0		5	
1			

mit Werten

C #

```
int[,] myArray = { {1, 2, 3}, {4, 5, 6} }; // 3 Zeilen  
                                              // 2 Spalten  
Console.WriteLine (myArray[0, 1]);
```

	0	1	2
1	1	2	3
2	4	5	6

# Übungen

Uebungen\05-Datenstrukturen\Übung Array.pdf

Uebungen\ 05-Datenstrukturen\Übung König und seine Gefangenen.pdf



Software-Lösungen. Einfach. Clever.



# Auflistungen

Listen



Software-Lösungen. Einfach. Clever.



# Listen

## variable Anzahl Werte

- keine feste Länge
- Grösse passt sich dynamisch an
- Unterscheidung identischer/unterschiedlicher Datentyp

# List

## Listen für identische Datentypen

Nur Daten des gleichen Typs

Deklaration einer Liste vom Typ *int* mit **List**

C #

```
List<int> meineListe = new List<int>();
```

Hinzufügen von Werten

C #

```
meineListe.Add(12);
```

Entfernen von Werten

C #

```
meineListe.Remove(12);  
// oder  
meineListe.RemoveAt(0);
```

# ArrayList

## Listen für unterschiedliche Datentypen

Daten unterschiedlichen Typs

Deklaration einer **ArrayList**

C #

```
ArrayList meineListe = new ArrayList();
```

Hinzufügen von Werten

C #

```
meineListe.Add("Hallo");  
meineListe.Add(25);
```

Entfernen von Werten

C #

```
meineListe.Remove("Hallo");  
// oder  
meineListe.RemoveAt(0);
```



# Übungen

Übungsaufgaben 5.6 s.60



Software-Lösungen. Einfach. Clever.