

Arrays und Bedingungen

Arrays

- Aufzählung gleichartiger Datentypen

```
int[] zahlen = new int[] { 2, 4, 6, 8, 10 }; //Länge = 5 / max. Index = 4  
string[] buchstaben = new string[] { "A", "B", "C" }; //Länge = 3 / max. Index = 2  
double[] kommazahlen = new double[10]; //Länge = 10 / max. Index = 9
```

- Zugriff erfolgt über nullbasierenden Index

```
zahlen[2]; // 6  
buchstaben[1]; // "B"  
buchstaben[2] = "Z"; // { "A", "B", "Z" }
```

Funktionen und Eigenschaften von Arrays

```
int[] zahlen = new int[] { 2, 9, 5 };
```

Funktionsname	Beschreibung	Beispiel
Length	Gibt die Anzahl der Elemente im Array zurück	zahlen.Length; //3
Contains(value)	Prüft ob Array ein Element enthält	zahlen.Contains(5); // true
Max()	Gibt den größten Wert im Array zurück	zahlen.Max(); //9
Min()	Gibt den niedrigsten Wert im Array zurück	zahlen.Min(); //2
First()	Gibt das erste Element im Array zurück	zahlen.First(); //2
Last()	Gibt das letzte Element im Array zurück	zahlen.Last(); //5
Sum()	Gibt die Summe aller Elemente zurück	zahlen.Sum(); //16

Mehrdimensionale Arrays

- Ein Array kann sich über mehrere Dimensionen erstrecken

```
//Deklaration eines mehrdimensionalen Arrays
```

```
int[,] zweiDimensionen;
```

```
//zweidimensionales Array mit zwei Zeilen und fuenf Spalten
```

```
zweiDimensionen = new int[,] { { 1, 2, 3, 4, 5 }, { 6, 7, 8, 9, 10 } };
```

```
//zweidimensionales Array mit vier Zeilen und drei Spalten
```

```
zweiDimensionen = new int[4, 3];
```

```
//Zugriff über den Index
```

```
int wert = zweiDimensionen[1, 2];
```

```
//Zuweisung über den Index
```

```
zweiDimensionen[2, 1] = 5;
```

Funktionen und Eigenschaften von mehrdimensionalen Arrays

```
int[, ,] array3D;  
array3D = new int[2, 3, 2] { { { 1, 2 }, { 3, 4 }, { 5, 6 } }, { { 7, 8 }, { 9, 10 }, { 11, 12 } } };
```

Funktionsname	Beschreibung	Beispiel
Length	Gibt die Anzahl aller Elemente im Array zurück	array3D.Length; // 12
Rank	Gibt die Anzahl der Dimensionen zurück	array3D.Rank; // 3
GetLength(int)	Gibt die Länge der jeweiligen Dimension zurück	array3D.GetLength(0); // 2 array3D.GetLength(1); // 3 array3D.GetLength(2); // 2

Boolesche Logik

- Jede Operation ergibt entweder **true** oder **false**

```
a == b;    //a gleich b
a != b;    //a ungleich b
a < b;     //a kleiner b
a > b;     //a größer b
a <= b;    //a kleiner oder gleich b
a && b;    //a und b sind wahr
a || b;    //a oder b ist wahr
a ^ b;     //entweder a oder b
!a;        //true wird zu false, false zu true
```

Bedingungen

```
if (Bedingung A)
{
    //Anweisungen wenn Bedingung A wahr
}
else if (Bedingung B)
{
    //Anweisungen wenn Bedingung A falsch und B wahr
}
else
{
    //Anweisungen wenn Bedingung A und B falsch sind
}
```