



colourbox.de

Fachklasse 'Rechteck'

Inhalt

Typische Bestandteile von Fachklassen	2
Attribute	2
Konstruktoren	2
Properties	2
Methoden	2
Klassendiagramm der Klasse "Rechteck"	3
Logische Skizze des Arbeitsspeichers bei der Erstellung eines Rechteck-Objekts	3
Definition der Fachklasse 'Rechteck'	4
Anwendungsprogramm für die Klasse 'Rechteck'	5
Ausgabe auf die Konsole	5

Typische Bestandteile von Fachklassen

Attribute

Attribute sind die Datenbestandteile einer Klasse.

In der Regel werden Attribute mit dem Zugriffsrecht "private" definiert. Dadurch sind sie nur von innerhalb der Klasse direkt ansprechbar und vor unbeabsichtigter Änderung im Anwendungsprogramm geschützt (Prinzip der "Datenkapselung").

Konstruktoren

Konstruktoren sind spezielle Methoden, die bei der Erstellung von neuen Objekten aufgerufen werden. Konstruktoren haben die Aufgabe, die Attribute mit sinnvollen Anfangswerten zu versehen. Dafür werden ihnen häufig entsprechende Parameter mitgegeben.

Konstruktoren haben in der Regel das Zugriffsrecht "public", sodass auch im Anwendungsprogramm Objekte von der Klasse erstellt werden können.

Properties

Properties erlauben den kontrollierten Zugriff auf die Werte der Attribute. Dieser Zugriff kann nach Wahl lesend oder auch schreibend sein ("get"- bzw. "set"-Properties). Die Syntax der Verwendung einer Property entspricht der Verwendung eines Attributs. Properties können bei Bedarf prüfenden oder berechnenden Code enthalten.

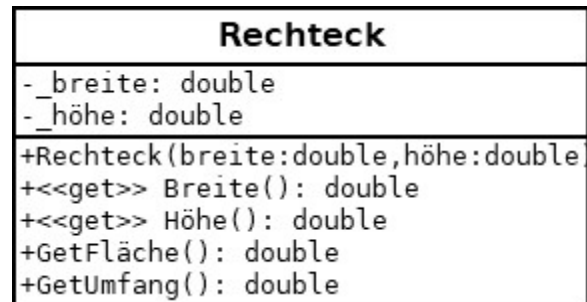
Properties haben häufig das Zugriffsrecht "public".

Methoden

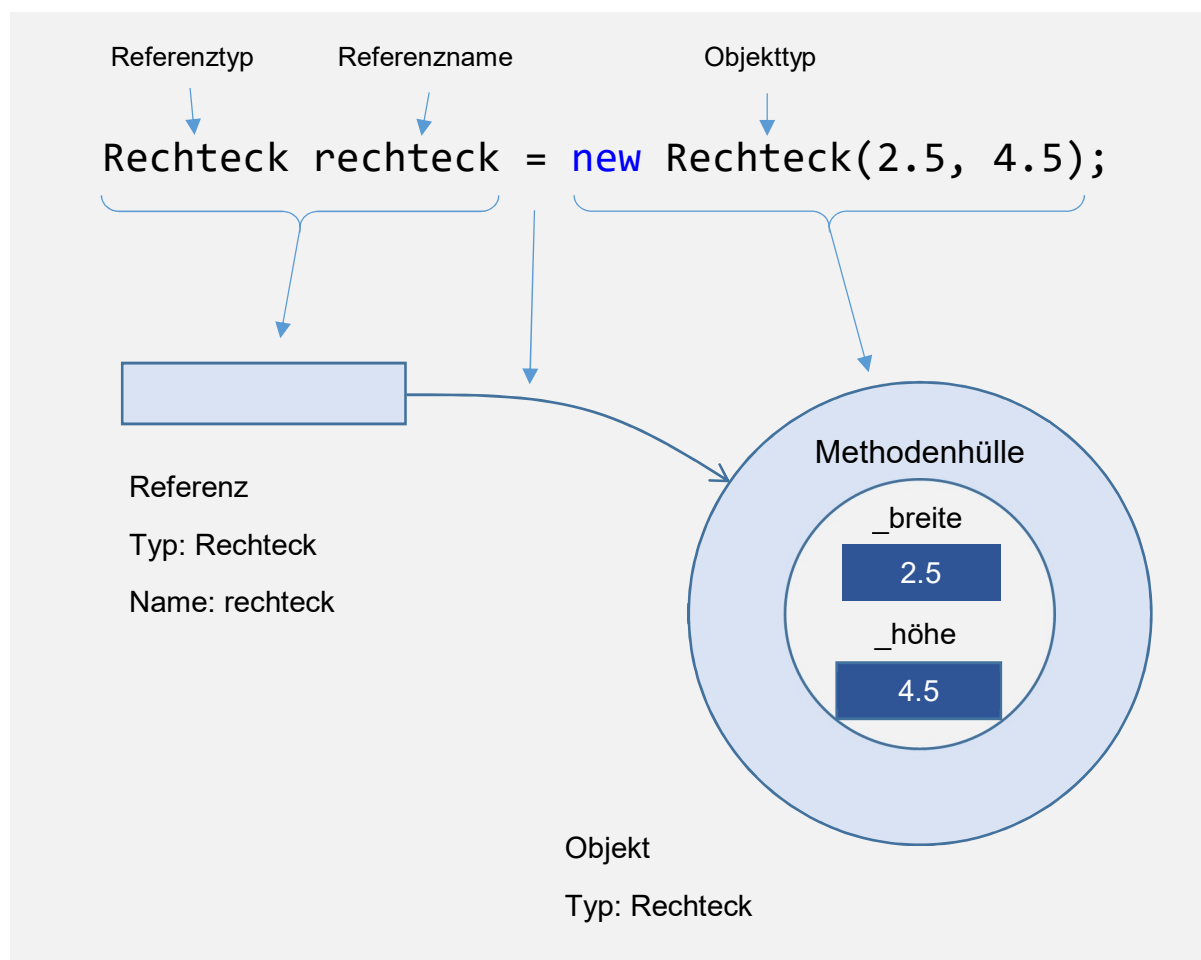
Methoden enthalten Algorithmen, mit denen aus den Attributen Werte berechnet oder die Werte der Attribute verändert werden können.

Methoden haben häufig das Zugriffsrecht "public".

Klassendiagramm der Klasse "Rechteck"



Logische Skizze des Arbeitsspeichers bei der Erstellung eines Rechteck-Objekts



Definition der Fachklasse 'Rechteck'

```
using System;

namespace RechteckTest
{
    // Fachklasse 'Rechteck'.
    public class Rechteck
    {
        // Attribute.
        private double _breite;
        private double _höhe;

        // Konstruktor.
        public Rechteck(double breite, double höhe)
        {
            _breite = breite;
            _höhe = höhe;
        }

        // get-Properties.
        public double Breite
        {
            get
            {
                return _breite;
            }
        }

        public double Höhe
        {
            get
            {
                return _höhe;
            }
        }

        // Rechenmethoden.
        public double GetFläche()
        {
            return _breite * _höhe;
        }

        public double GetUmfang()
        {
            return 2.0 * (_breite + _höhe);
        }
    }
}
```

Anwendungsprogramm für die Klasse 'Rechteck'

```
using System;

namespace RechteckTest
{
    //
    // Demonstrationsprogramm zur Verwendung der Fachklasse 'Rechteck'.
    //
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Kenngrößen eines Rechtecks.");
            //
            // Ein Rechteck-Objekt erstellen und mit Hilfe einer Referenz verwalten.
            //
            Rechteck rechteck = new Rechteck(2.5, 4.5);
            //
            // Kenngrößen abrufen und ausgeben.
            //
            // Aufruf von Properties.
            //
            Console.WriteLine("Breite: " + rechteck.Breite);
            Console.WriteLine("Höhe : " + rechteck.Höhe);
            //
            // Aufruf von Methoden.
            //
            Console.WriteLine("Fläche: " + rechteck.GetFläche());
            Console.WriteLine("Umfang: " + rechteck.GetUmfang());
        }
    }
}
```

Ausgabe auf die Konsole

```
Kenngrößen eines Rechtecks.
Breite: 2,5
Höhe : 4,5
Fläche: 11,25
Umfang: 14
```