Komposition

Die Komposition stellt die stärkste Form einer Objektbeziehung dar und ist eine Beziehung vom Typ „**has-a**“ („**hat-ein**“) oder „**is-part-of**“ („**ist-Teil-von**“); je nach Sichtweise, ausgehend vom Ganzen oder ausgehend vom Teil. Das Ganze wird auch „Kompositum“ genannt.

Für eine Komposition gilt:

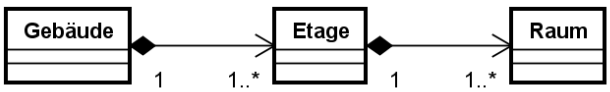
1. Eine Komponente ist in ihrer Existenz abhängig vom Kompositum!
   Die Lebensdauer des Kompositums bestimmt auch die Lebensdauer der Komponenten.
2. Eine Komponente gehört genau zu einem Kompositum!
   Eine Komponente kann nicht mehreren Komposita zugeordnet sein.

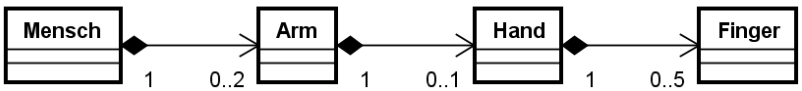
Eine Komposition wird wie im nachfolgenden Bild zu sehen ist, dargestellt. Auf der Seite des Ganzen (Kompositum) befindet sich eine ausgefüllte Raute.

Das Kompositum beinhaltet keine, ein oder mehrere Komponenten. Hingegen kann die Multiplizität auf der Seite des Kompositums nur „1“ sein. Zwar ist auf der Seite des Kompositums auch die Multiplizität „0..1“ denkbar, was ausdrückt, dass das Teil auch eine „kurze“ Zeit unabhängig vom Ganzen existieren kann, aber in der Regel ist die Multiplizität **genau** „1“!

Allerdings darf eine Komponente zur Lebenszeit entfernt und dann genau einem anderen Kompositum zugeordnet werden!

Die beiden Beispiele zeigen eine Komposition:





Ein Gebäude besteht aus mindestens einer Etage und eine Etage besteht aus mindestens einem Raum. Wird eine bestimmte Etage abgerissen, sind auch alle darin befindlichen Räume weg. Wird hingegen das ganze Gebäude abgerissen, sind alle Etagen mit allen darin befindlichen Räumen zerstört. Jede konkrete Etage gehört zu einem ganz bestimmten Gebäude und jeder konkrete Raum zu einer ganz bestimmten Etage.

Ein Mensch hat maximal zwei Arme, ein ganz bestimmter Arm kann nicht zu verschiedenen Menschen gehören. Jeder Arm hat maximal eine Hand, die wiederum maximal fünf Finger aufweist. Sowohl die Finger gehören zu einer ganz bestimmten Hand wie auch eine Hand zu genau einem Arm gehört.

Die Navigierbarkeit ist in der Regel beidseitig. Es macht keinen Sinn, dass das Kompositum seine Komponenten nicht kennt. Daher enthält das Assoziationsende auf der Komponentenseite einen Pfeil. Und umgekehrt kennt die Komponente üblicherweise auch das Ganze, zu dem sie gehört. Der Pfeil wird hier allerdings durch die Raute „verdeckt“.

Eine Implementation einer Komposition kann durch eine geschachtelte Klassendefinition erfolgen, wodurch die Existenzabhängigkeit besonders deutlich wird. Hierbei ist die Komponentenklasse in der Kompositumsklasse enthalten.

Allerdings können die Klassendefinitionen auch separat voneinander existieren, wobei die Objekterzeugung der Komponenten im Konstruktor des Kompositums durchgeführt wird.

Implementation einer Komposition

Nachfolgend ist das vorherige Beispiel mit den Klassen **`Mensch`**, **`Arm`** und **`Hand`** (unter Verzicht auf die Klasse **`Finger`**) in einer Minimalversion umgesetzt.

In der Definition der Klasse **`Arm`** (Datei **`Arm.cs`**) befindet sich eingebettet die Definition der Klasse **`Hand`**. Die Klasse **`Mensch`** hingegen ist in einer eigenen Datei **`Mensch.cs`** definiert.

Allen drei Klassen wurden Destruktoren hinzugefügt, um eine Ausgabe im Rahmen der Objektzerstörung durchzuführen. **Diese werden normalerweise nicht benötigt!**

```
using System;

namespace Komposition
{
    // Kompositumsklasse
    class Mensch
    {
        string name;
        Arm rechterArm;
        Arm linkerArm;

        public string Name
        {
            get { return name; }
            set { name = value; }
        }

        public Arm RechterArm
        {
            get { return rechterArm; }
            set { rechterArm = value; }
        }

        public Arm LinkerArm
        {
            get { return linkerArm; }
            set { linkerArm = value; }
        }

        public Mensch(string n)
        {
            name = n;
            Console.WriteLine(
                    "Mensch erstellt!");
            rechterArm = new Arm(this, true);
            linkerArm = new Arm(this, false);
        }

        ~Mensch()
        {
            Console.WriteLine(
                    "Mensch gelöscht!");
        }
    }
}
```

```
using System;

namespace Komposition
{
    class Arm
    {
        Mensch mensch;
        // true --> rechter Arm, false --> linker Arm
        bool armseite;
        Hand hand;

        public bool Armseite
        {
            get { return armseite; }
            set { armseite = value; }
        }

        public Arm(Mensch m, bool seite)
        {
            mensch = m;
            Armseite = seite;
            if (armseite)
                Console.WriteLine("rechter Arm erstellt!");
            else
                Console.WriteLine("linker Arm erstellt!");

            // Komponente wird erzeugt
            hand =  new Hand(this);
        }

        ~Arm()
        {
            if (armseite)
                Console.WriteLine("rechter Arm gelöscht!");
            else
                Console.WriteLine("linker Arm gelöscht!");
        }

        // Komponentenklasse Hand
        // Wenn das Kompositum "Arm" untergeht, wird auch
        // die Komponente "Hand" zerstört.
        class Hand
        {
            Arm arm;
            bool handseite;
            public Hand(Arm a)
            {
                arm = a;
                handseite = a.Armseite;
                if (handseite)
                    Console.WriteLine("rechte Hand erstellt!");
                else
                    Console.WriteLine("linke Hand erstellt!");
            }
            ~Hand()
            {
                if (handseite)
                    Console.WriteLine("rechte Hand gelöscht!");
                else
                    Console.WriteLine("linker Hand gelöscht!");
            }
        }
    }
}
```

Aufgabe

Erstellen Sie ein neues Projekt zu der eingangs dargestellten Komposition „Gebäude-Etage-Raum“.

Das Gebäude soll zwei Etagen mit jeweils fünf Räumen haben.
Jede Etage und jeder Raum haben eine Bezeichnung; z.B. „Etage 0“ und „Raum 1“.
Implementieren Sie die Komposition in der Form wie zuvor dargestellt: Gebäude als eigene Klasse, Etage und Raum verschachtelt.