

Einführung in die Objektorientierte Programmierung

Allgemeines

Objektorientierung in der Programmierung besteht nicht nur aus der Erstellung von Programmen, sondern ist viel weiter gefasst. Er betont besonders stark das Modellieren vor der Programmierung. Man unterscheidet drei Phasen:

1. **Objektorientierte Analyse (OOA):** Die reale Welt wird auf die Existenz von Objekten und Objektbeziehungen untersucht. Es wird ein objektorientiertes Modell der realen Welt erstellt und geprüft, welche Funktionalität es wem zur Verfügung stellen soll. Es wird gefragt, **was** mit den Objekten **warum** geschieht.
2. **Objektorientiertes Design (OOD) oder Entwurf:** Das objektorientierte Modell der realen Welt wird auf die Software übertragen und aufgrund von programmiertechnischen Notwendigkeiten ergänzt und verändert. Bei diesem Modell der Software-Architektur wird festgelegt, **wie** etwas geschehen soll.
3. **Objektorientiertes Programmieren (OOP) oder Implementierung:** Die Software-Architektur wird in ein lauffähiges Programm umgesetzt. Objektzustände werden durch Datenstrukturen dargestellt, Objektfunktionalität durch Algorithmen realisiert.

Aufgabe:

- a) Was versteht man unter einem Modell? Welchen Zweck hat es?
- b) Wie kann man die Qualität eines Modells beurteilen?

Begriffe

Objekt:

auch Instanz, Instance, Exemplar oder Repräsentant genannt.

Ein Objekt ist eine gedankliche oder reale Einheit in der Umwelt oder in der Software. Es ist konkret vorhanden und besitzt einen gewissen Zustand und ein bestimmtes Verhalten.

Operationen:

auch Methoden oder Dienstleistungen genannt.

Operationen sind Handlungen eines Objekts, die von ihm mit Hilfe einer Nachricht angefordert und dann ausgeführt werden. Operationen werden durch Methoden implementiert.

Eigenschaften:

auch als Datenelemente oder Datenfelder bezeichnet.

Die Bedeutung von Eigenschaften in der Objektorientierten Programmierung entspricht der aus dem Alltag. Eigenschaften beschreiben den Zustand eines Objektes

Bsp.: Das Objekt Apfel hat die Eigenschaften Farbe, Geschmack, Masse.

Klasse:

Die Klasse ist die Definition der Eigenschaften und Operationen für eine bestimmte Menge von Objekten. Alle Objekte dieser Klasse entsprechen dieser Definition. Eine Klasse ist ein "Bauplan für Objekte". Ein Objekt hat genau die Eigenschaften und Operationen, die in dieser Klasse definiert worden sind. Jedes Objekt hat aber individuelle Werte der Eigenschaften.

Nachrichten:

Auch Messages oder Botschaften genannt. Nachrichten sind ein Mechanismus zur Kommunikation von Objekten untereinander.

Bsp.: Haus

Ein Haus hat eine Anschrift, ein Dach, Wände, eine Tür, eine Farbe. Ein Objekt dieser Klasse Haus ist dann ein ganz bestimmtes Haus in einer bestimmten Straße mit einer bestimmten Hausnummer.

z. B. Ein Objekt der Klasse Haus ist ein gelbes Haus in der Hauptstraße 4 mit Holzwänden. Ein anderes Objekt der Klasse Haus ist ein Backsteinhaus in der Bahnhofstraße 2.

Kapselung:

Als Kapselung bezeichnet man die Zusammenfassung von Name, Zustand und Verhalten eines Objekts.

Geheimnisprinzip:

Das Geheimnisprinzip bezeichnet die Unsichtbarkeit von Eigenschaften und Methoden einer Klasse oder eines Objekts für andere Elemente des Objektmodells.

Dies bedeutet, dass der Benutzer einer Klasse nur soviel sehen darf, wie für die Anwendung und die Arbeit notwendig ist. Deshalb wird die Klasse in verschiedene Sektionen der Sichtbarkeit eingeteilt.

UML:

UML steht für Unified Modelling Language und ist der Standard zur Spezifizierung, Visualisierung und Konstruktion von Softwaresystemen. Dabei stellt UML eine Notation zur Beschreibung objektorientierter Softwaresysteme dar und verfolgt das Ziel, eine universelle Beschreibungssprache für alle Arten von objektorientierten Softwaresystemen zur Verfügung zu stellen, die von den Entwicklern wie von den Auftraggebern verstanden wird. UML ermöglicht die Erstellung fertiger Softwaresysteme auf dem Papier und fördert die Entwicklung modularer Programmierung.

In diesem Kurs werden wir uns v.a. auf so genannte *Klassendiagramme* konzentrieren.

Bsp.: Klasse Haus in UML

| | |
|--|--------------------------|
| Haus | Name der Klasse |
| Farbe: Text Anschrift: Text Wand: Wandklasse Tür: Türklasse | Eigenschaften der Klasse |
| void ÖffneTür(); void SchließeTür(); void ÖffneFenster(); void SchließeFenster(); | Operationen der Klasse |

Eine Objekt der Klasse Haus:

| |
|---|
| <u>Weißes Haus: Haus</u> |
| Farbe = „weiß“ Anschrift = „Pennsylvania Ave 1600“ |

Beziehungen zwischen Objekten oder Klassen werden durch  dargestellt.

Aufgaben:

1. Modelliere eine Klasse Auto und teste die Modellierung mittels verschiedener Objekte der Klasse Auto.
2. Überlege, aus welchen anderen Objekten ein Auto aufgebaut ist. Erstelle die Klasse und stelle deren Struktur untereinander dar.