

Objekt (objektorientierte Programmierung)

aus Wikipedia, der freien Enzyklopädie

Ein **Objekt** ist in der objektorientierten Programmierung eine konkrete Ausprägung einer Klasse (Man sagt auch *Exemplar* oder *Instanz* einer Klasse). Das Objekt definiert sich über seine Identität, seinen Zustand und sein Verhalten (Bonmot: "*Ist was, hat was, kann was*").

- Jedes Objekt besitzt eine **Identität**, die es einzigartig in der Menge von Objekten eines Systems macht.
- Der **Zustand** eines Objekts wird durch die Werte der Attribute seiner Klasse bestimmt.
- Das **Verhalten** eines Objekts ergibt sich aus den zur Verfügung stehenden Methoden der zugrundeliegenden Klasse.

Man kann sich das Erstellen von Objekten nach einer Klasse vorstellen wie das Fertigen von Autos aus dem Konstruktionsplan eines bestimmten Fahrzeugtyps. Jedes erzeugte Auto für sich genommen hat eine einzigartige Identität, denn alle Autos können unabhängig voneinander betrieben werden. Klassen sind die Konstruktionspläne für Objekte.

In objektorientierten Programmen bilden Objekte die Einheiten der Datenkapselung. Im Normalfall dienen sie dazu, Dinge der realen Welt zu modellieren (Abstraktion).

In den meisten objektorientierten Sprachen werden alle Objekte nach dem Konstruktionsplan von Klassen erzeugt. Durch die Definition einer Klasse sind die Merkmale all derjenigen Objekte festgelegt, die aus dieser einen Klasse hervorgehen können. Objekte, die aus von einer Basisklasse abgeleiteten Klassen erzeugt wurden, können genauso wie Objekte dieser Basisklasse verwendet werden.

In rein objektorientierten Sprachen wie Smalltalk werden, dem Prinzip *alles ist ein Objekt* folgend, auch elementare Typen wie Ganzzahlen (Integer) durch Objekte repräsentiert. Auch Klassen selbst sind hier Objekte, die wiederum Ausprägungen von Metaklassen sind. Viele Sprachen, unter anderem C++ und Java folgen allerdings nicht der reinen Lehre der Objektorientierung; daher sind dort elementare Typen keine vollwertigen Objekte, sondern müssen auf Methoden und Struktur komplett verzichten.

In einigen objektorientierten Programmiersprachen wie zum Beispiel JavaScript, NewtonScript und Self wird auf die Deklaration von Klassen gänzlich verzichtet. Statt dessen werden bestehende Objekte, so genannte Prototypen abgeleitet. Die Attribute und Methoden des Prototypen kommen immer dann zum Einsatz, wenn sie nicht im abgeleiteten Objekt explizit überschrieben wurden. Dies ist vor allem für die Entwicklung kleinerer Programme von Vorteil, da es einfacher und zeitsparend ist.

Weblinks

- Prototypen in der Objektorientierten Programmierung (englisch)
(<http://lieber.www.media.mit.edu/people/lieber/Lieberary/OOP/Delegation/Delegation.html>)

Von "http://de.wikipedia.org/wiki/Objekt_%28objektorientierte_Programmierung%29"

Eingeordnet unter: Objektorientierte Programmierung

- Impressum | Diese Seite wurde zuletzt geändert um 22:02, 4. Sep 2004.
- Der Inhalt dieser Seite steht unter der GNU Free Documentation License.