

Funktionen

Eine Funktion erledigt eine abgeschlossene Teilaufgabe. Die notwendigen Daten werden der Funktion mitgegeben. Das Ergebnis der erledigten Aufgabe gibt sie an den Aufrufer zurück. Eine Funktion muss nur einmal definiert werden. Anschließend kann sie beliebig oft nur durch Nennung ihres Namens aufgerufen werden, um die ihr zugewiesene Teilaufgabe abzuarbeiten. Bereits vorhandene Standardlösungen von Teilaufgaben können aus Funktionsbibliotheken abgerufen werden – ebenso wie neu entwickelte Funktionen in Bibliotheken aufgenommen werden können.

Quelle : (gekürzt) Der C++ Pogrammierer / Ulrich Breyman / Hanser- Verlag / 5.Auflage

1. Funktionsdefinition und Aufruf

Funktionen müssen stets vor ihrem Aufruf (hier in ⑧) definiert werden (hier in ①).

Methoden geben genau einen oder keinen Rückgabewert (mit **return**) zurück. Der Rückgabewert wird vor dem Methodennamen angegeben. Hinter dem Methodennamen folgt die Parameterliste. Eine Parameterliste wird immer in runden Klammern angegeben. Wenn keine Parameterliste angegeben wird, müssen die runden Klammern (dann leer) geschrieben werden.

①	[Zugriffsmod.][static] Rückgabewert Bezeichner ([Typ Arg ₁ [,Typ Arg ₂ ...]]) {
②	[ggf. Deklaration lokaler Variablen]
③	Anweisungen;
④	:
⑤	return Rückgabewert;
⑥	}
⑧	[Variable =] Bezeichner ([Wert ₁ [, Wert ₂ ...]]);

Deklaration und Aufruf von Methoden (Java-Unterprogramme)

2. Funktionen überladen

In Java werden Funktionen nicht nur anhand ihres Namens, sondern auch an der Parametersignatur unterschieden. Es kann also Funktionen gleichen Namens geben, wenn sie sich in Art und/oder Anzahl der Parameter unterscheiden (der Rückgabewert gehört nicht zur Parametersignatur).

Funktionsaufruf		überladene Funktionen	
①	<code>gibAus();</code>	Aufruf ① ruft Funktion ① auf <input checked="" type="checkbox"/>	① <code>void gibAus() { // Funktionsrumpf }</code>
②	<code>gibAus(20);</code>	Aufruf ② ruft Funktion ② auf <input checked="" type="checkbox"/>	② <code>void gibAus(int a) {...}</code>
③	<code>gibAus(20, 4.5);</code>	Aufruf ③ ruft Funktion ③ auf <input checked="" type="checkbox"/>	③ <code>void gibAus(int a, float b) {...}</code>

3. Parameterübergabe an Funktionen

Die Übergabe der Parameter erfolgt grundsätzlich **ByValue**. Das heißt, dem Unterprogramm wird eine Kopie der Originalvariablen übergeben. Änderungen an der übergebenen Variable im Unterprogramm **wirken sich nicht** auf das Hauptprogramm aus.

Auch bei Referenztypen (Objekten) wird eine Kopie der Referenzvariable übergeben. Da Original und Kopie aber auf dasselbe Objekt verweisen (und nicht das Objekt „transportiert“ wird), **wirken sich** Änderungen am Objekt innerhalb des Unterprogramms auch **auf das Originalobjekt aus**. Diese Übergabeart bezeichnet man **ByReference**.