

Einführung in die Programmierung für NF

Übung 03

06.11.2013

Inhalt

- Methoden in Java
- Gültigkeitsbereiche
- EBNF

Methoden in Java

- In Methoden werden Ausdrücke und Anweisungen zusammengefasst
- Methoden können dabei Übergabeparameter bekommen, die in der Methode verwendet werden
- In Java wird dabei der Datentyp angegeben

Sichtbarkeit

Rückgabewert

Funktionsname

Parameter

```
public static void functionXY (int x){...}
```

Übergabeparameter

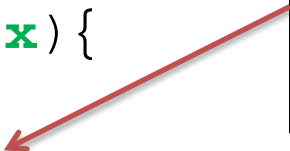
- Übergabeparameter sind optional, gibt es keine, werden leere Klammern geschrieben
- Übergabeparameter sind nur innerhalb ihrer Funktion sichtbar
- Wenn Übergabeparameter und Klassenvariable den gleichen Namen haben, kann mit „`this`“ auf die Klassenvariable zugegriffen werden

Zugriff auf Klassenvariablen

- Beispiel

```
public class Testklasse {  
    int x = 5;  
    int y = 3;  
  
    public int someSum(int x) {  
  
        return x + this.x + y;  
  
    }  
}
```

Hier könnte auch `this.y` stehen, muss aber nicht, da nur `y` bereits eindeutig ist



Rückgabewerte

- Wie bei mathematischen Funktionen, können Funktionen einen Rückgabewert haben
- Funktionen ohne Rückgabewert haben in Java den Rückgabewert „void“

```
public int fun1 () {...}  
public boolean fun2 () {...}  
public void fun3 () {...}
```

Rückgabe

- Die Rückgabe einer Funktion in Java erfolgt mit dem Befehl „`return`“
- Mit einem Aufruf von `return` endet die Funktion, auch wenn noch weiterer Inhalt folgt – dieser wird dann nicht ausgeführt
- Funktionen ohne Rückgabewert (`void`) haben keinen `return`-Befehl

Rückgabe

- Beispiel

```
public class Testklasse2 {  
  
    public int someStuff(int x){  
  
        if(x > 2){  
            return 0;  
        }  
  
        return x - 17;  
  
    }  
}
```

Wenn x größer als 2 ist,
wird mit der Rückgabe
von 0 die Funktion
beendet

Methodenaufruf

- Der Aufruf einer Methode muss immer mit allen Parametern erfolgen
- Ergebnistyp der Funktion und Verwendung müssen natürlich übereinstimmen

Fragen zu Methoden?

Gültigkeitsbereiche

- Der Gültigkeitsbereich einer Variable ist der umfassende Block, ab dem Zeitpunkt der Deklaration
- Beispiele
 - Variablen in Methoden oder Schleifen sind außerhalb nicht sichtbar, zuvor deklarierte Variablen innerhalb der Methode oder Schleife aber schon

EBNF

- Erweiterte Backus-Naur-Form
- Sehr gute Übersicht unter http://de.wikipedia.org/wiki/Erweiterte_Backus-Naur-Form
- Die BNF ist eine Notation, um die Syntax eines Ausdrucks zu definieren, also was erlaubt ist im Ausdruck, und was nicht

EBNF - Syntax

- Die EBNF beschreibt immer einen Text, der am Ende ausgegeben werden kann
- Zeichen, die am Ende tatsächlich ausgegeben werden (Terminalsymbole), stehen in Anführungszeichen " "
- Alle weiteren Symbole sind Platzhalter (Variablen) für weitere Textmöglichkeiten

EBNF - Syntax

- Zuweisungen erfolgen mit =
- Beispiel

IrgendeineZiffer = „5“

EBNF - Syntax

- Alternativen werden mit einem einfachen | notiert („oder“)
- Beispiel

```
ZifferAußerNull = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;  
Ziffer = "0" | ZifferAußerNull ;
```

EBNF - Syntax

- Reihungen werden mit `,` notiert („und“)
- Beispiel

Zwoelf = "1", "2" ;

Zweihundertundeins = "2", "0", "1" ;

Dreihundertzwoelf = "3", Zwoelf ;

ZwoelfTausendzweihunderteins = Zwoelf, Zweihundertundeins ;

EBNF - Syntax

- Optionale wiederholbare Ausdrücke („kein mal oder beliebig oft“) werden in geschweifte Klammern geschrieben { }
- Beispiel

NatuerlicheZahl = ZifferAußerNull, { Ziffer } ;

Hier passen die Texte *1, 2, ...,10,...,12345, ...* . Zu beachten ist, dass alles, was innerhalb der geschweiften Klammern steht, beliebig oft, jedoch auch *keinmal* vorkommen kann.

EBNF - Syntax

- Optionale Ausdrücke („entweder da oder nicht“) werden in eckige Klammern geschrieben []
- Beispiel

GanzeZahl = "0" | ["-"], NatuerlicheZahl ;

Eine ganze Zahl ist also die Null (0) oder eine natürliche Zahl, der optional ein Minuszeichen vorangestellt werden kann. Hier passen also alle ganzen Zahlen wie 0, -3, 1234 etc.

EBNF - Syntax

- Feste Anzahl an Wiederholungen („x mal“) werden mit x^* geschrieben
- Beispiel

Viermalvier = $4^* "4"$;

EBNF - Syntaxübersicht

Definition	=
Aufzählung	,
Endezeichen	;
Alternative	
Option	[...]
Optionale Wiederholung	{ ... }
Gruppierung	(...)
Anführungszeichen	" ... "

EBNF – Beispiel Backrezept

- Beschreibung von Zutaten in einem Rezept
- Dazu werden folgende Angaben gemacht:
 - Anzahl der Teelöffel („TL“) oder Esslöffel („EL“), die eine positive ganze Zahl ungleich 0 ist
 - Um welche Zutat es sich handelt, vorhanden sind „Salz“, „Zucker“, „Mehl“ und „Milch“
- Ein Rezept besteht aus beliebig vielen, aber mindestens einer Mengenangabe

EBNF – Beispiel Backrezept

- Beispiele für gültige Rezepte
 - 5 EL Zucker 10 EL Mehl 20 TL Milch
 - 1 TL Salz 1 TL Salz
- Ungültig sind z.B. folgende Konstruktionen
 - 0 TL Salz
 - 5 TL 3 EL Milch
 - 10 EL Mehl Zucker

EBNF – Beispiel Backrezept

- Das Startsymbol ist **Backrezept**
- Folgende Definitionen sind bereits gegeben:

Zahl = ZifferAußerNull, {Ziffer}

ZifferAußerNull = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;

Ziffer = "0" | ZifferAußerNull ;

- Los geht's!

EBNF – Ableitungen

- Ableitung mit der erstellten Grammatik von folgendem Ausdruck:

1 TL Zucker 3 EL Mehl

Fragen zum Übungsblatt?