**Kapitel 3**

Datentypen

Bool’sche Typen Wahrheitswerte (True/False)

Numerische Typen Byte, Short, Integer, Long Menge ganzer Zahlen

Double, Float Menge der reellen Zahlen

Alphanumerisch Char Menge vereinbarter Zeichen

Komplexe Daten String Verbindung von Charactern

Array Definition eines n-Dimensionalen Feldes

Referenzdatentypen individuell definierte Typen

Deklaration von Variablen

werden mit Datentyp und Variablennamen deklariert

*Typname Variablenname;*

Deklaration von Konstanten

Werden „Final“, Datentyp und Variablennamen deklariert

*Final Typname Konstantenname;*

Literale

Sind fest definierte Zeichenfolgen zur Darstellung von Werten für Basis-Datentypen.

Numerische Literale

Ganzzahlen

Dezimal Literale Ziffern 0-9

Oktal Literale Präfix 0; Ziffern 0-7

Hexadezimal Literale Präfix 0x; Ziffern 0-9; Buchstaben a-f;

Durch Suffix L/l wird ein Literal vom Typ Long erzeugt

Fließkommazahlen

Float und Double Vorkommateil, Dezimalpunkt, Nachkommateil, Exponent, Suffix

Unterscheidung von Float und Double durch Suffix f/d

Exponent wird durch e/E eingeleitet

Vor- oder Nachkomma stelle kann ausgelassen werden

Exponent und Suffix sind optional

Ohne Suffix immer Double

Alphanumerisch

Boolean true und false

Char In Hochkommata setzen

String In doppelte Hochkommata setzen

Deklaration von Variablen

Int „Name“; boolean „Name“;

Initialisierung

„Name“ = „x“; „Name“ = true/false;

Kombination

Int “Name” = “x”;^ boolean “Name” = true/false;

Konstanten dasselbe mit “final” vorne dran

Arrays

- Mehrdimensionale Feldvariablen die aus Elementen desselben Datentyps bestehen

- Größe kann zur Laufzeit festgelegt werden, aber nicht mehr geändert

- Arrays sind in Java Objekte mit verschiedenen Methoden

- Elemente sind bei length (Datentyp integer) = n von 0 bis n-1 durchnummeriert

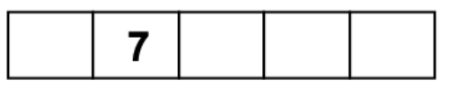
- Zugriff auf einzelne Elemente erfolgt über nummerischen Index

Deklaration von Arrays

- syntaktisch ähnlich der von Variablen

- Unterschied: an den Typ- oder Variablennamen werden [] gehängt

- die Initialisierung erfolgt über den new-Operator oder durch Zuweisung eines Array-Literale

Bsp.

int [] zahl = {1,2,3,4,5};

zahl [1] = 7;

Ein Bild, das Shoji, Kreuzworträtsel, Gebäude, Krabbe enthält.

Automatisch generierte Beschreibung

int [] [] matrix = new int [3] [3];

matrix [1] [2] = 15;

Ein Bild, das Shoji, Gebäude enthält.

Automatisch generierte Beschreibungint [] [] [] wuerfel = new int [3] [3] [3];

wuerfel [2] [] [] = 73;

Referenzdaten

- sind im Gegensatz zu einfachen Datentypen komplexe Typen

- Bsp. Arrays, Strings, Objekte, wobei Arrays und Strings besondere Objekte sind

- Bei Strings und Arrays kennt der Compiler Literale, die den Operator new überflüssig machen

- Initialisierung reicht nicht aus, müssen (außer Strings und Arrays) mit „new“ erzeugt werden

- sind Verweise auf Objekte