**Definition:**

* eine Variable ist ein Behälter für Werte(Zahlen, Zeichen, Texte, Objekte),
* sie hat einen *Namen*, einen **Datentyp** und einen Speicherplatz,
* der Speicherplatz wird vom System automatisch vergeben,
* Variablen werden deklariert, d.h. dem System bekannt gemacht,
* Variablen werden definiert, d.h. erstellt

**Datentypen**

|  |  |  |
| --- | --- | --- |
| Schlüsselwort(Datentyp) | Bedeutung(Gruppe) | Beispiel |
| **boolean** | Logische Werte | true, false |
| **char** | Zeichen | 3, A, $, ä, ĉ, æ, }, ~, ¿ |
| **short**  **int**  **long** | Ganzzahlen | -15, 0, 2, 210 |
| **float**  **double** | Gleitkommazahlen | -14.652, 0.0, 3.1415 |

**Variablennamen (inkl. Java-Konvention)**

* gültige Zeichen: A-Z, a-z, 0-9 und \_
* erstes Zeichen immer nur a-z; keine Zahl oder A-Z
* Groß- und Kleinschreibung ist von Bedeutung Summe ≠ summe ≠ SUMME
* keine Schlüsselwörter benutzen z.B. int, switch, for, class usw.
* der Name sollte selbsterklärend sein

**Variablendeklaration und Variablendefinition**

Möglichkeit 1

**Datentyp** *variablenname***;**

Möglichkeit 2

**Datentyp** *variablenname* = Startwert**;**

Möglichkeit 3 bei mehreren Variablen gleichen **Datentyps**

**Datentyp** *variablenname1***,** *variablenname2* = Startwert**,** *variablenname3* = Startwert**;**

**Beispiele**

**boolean** *wahr* = false;

**int** *integer*, *zahl*, *ganzzahl*;

**char** *zeichen*;

**char** *fragezeichen* = ‘?‘;

**float** komma = 14.75f; //Werte für den Datentyp float müssen mit f gekennzeichnet werden

**double** *zahl1* = 0.1, *zahl2*, *zahl3* = -16.125, *zahl4*, *zahl5*;

**Beispiele für Größe und Wertebereich von Datentypen:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Datentyp | Bedeutung |  | | |
| Größe (Byte) | kleinster Wert | größter Wert |
| boolean | Logische Werte | 1 Bit | true, false | |
| char | Zeichen | 2 | 0 | 65535 |
| byte | Ganzzahlen | 1 | -128 | 127 |
| short | 2 | -32 768 | 32 767 |
| int | 4 | -2 147 483 648 | 2 147 483 647 |
| long | 8 | -9,2E+18 | 9.2E+18 |
| float | Einfach genaue Gleitkommazahlen | 4 | -1.4E-45 | +3.4E+38 |
| 8 Dezimalstellen | |
| double | Doppelt genaue Gleitkommazahlen | 8 | -4.9E-324 | ±1.7E+308 |
| 15 Dezimalstellen | |
| String | Zeichenkette | - | | |

**Konstanten**

Werte, die im laufenden Programm nicht verändert werden sollen, können mit dem Schlüsselwort **final** als Konstanten definiert werden:

|  |
| --- |
| final double *PI* = 3.1415926535897932385; // Definiert die Konstante PI  final int *MWST* = 19; //Definiert die Konstante MWST |

Der Wert einer Konstante kann im Programm nicht mehr geändert werden.

**Eine Konstante muss also bei ihrer Definition mit einem Wert initialisiert werden.**

**Welcher Datentyp ist für den jeweiligen Verwendungszweck sinnvoll?**

**Wählen Sie bitte außerdem einen aussagekräftigen Variablennamen!**

|  |  |  |
| --- | --- | --- |
| **Verwendungszweck** | **Datentyp** | **Variablenname** |
| Das Alter einer Person (in Jahren). | short | alter\_person |
| Die Anzahl aller Sonnen einer Galaxie.  (< 100.000.000) | int | sonnen |
| Die jährliche Steigerung eines Aktienkurses in Prozent. | float | Aktionssteigerung |
| Das Gewicht einer Person in ganzen kg. | short | Gewicht\_kg |
| Der Buchstabe ‚j‘ oder ‚n‘. | char | J oder N |
| Die Bevölkerungszahl eines Landes. | int | bevoelkerungszahl |
| Die Mitarbeiterzahl eines mittelständigen Unternehmens. | int | Mitarbeiterzahl\_Unternehmen |
| Das durchschnittliche Jahreseinkommen eines Arbeitnehmers in Euro. | float | Jahreseinkommen\_Arbeitnehmer |
| Der Anfangsbuchstabe eines Namens. | char | Erste\_buchstabe |
| Die Fläche eines Grundstücks in ganzen Quadratmetern. | int | Grundstucks\_flaeche |
| Die Antwort eines Lügendetektors. | boolean | wahr |
| Ihren Kontostand. | short | kontostand |
| Für die Zahl π. | double | PI |

**Welche der folgenden Bezeichner sind gültig?**

**Wo könnte es welche Probleme geben?**

|  |  |  |
| --- | --- | --- |
| **Variablendeklaration** | **Konvention** | **Kommentar** |
| **int** *\_\_zaehler***;** | gültig ungültig X | Es fängt mit einem „ \_“ an |
| **float** *mwst\_1***;** | gültig X ungültig |  |
| **int** *3\_teilnehmer\_nr***;** | gültig ungültig X | Es fängt mit einer Zahl an |
| **double** *masse#***;** | gültig ungültig X | Es hat einen ungültigen Charakter (#) |
| **char** *wähler***;** | gültig ungültig X | Es hat einen ungültigen Charakter (ä) |
| **int** *telefon\_nr***;** | gültig X ungültig |  |
| **char** *bUcHsTaBe***;** | gültig X ungültig | Komisch, aber passt |
| **float** *summe\_gehalt***;** | gültig X ungültig |  |
| **long** *anzahl-telefone***;** | gültig ungültig X | Es hat einen ungültigen Charakter (-) |
| **int** *long***;** | gültig ungültig X | Ein Schlüsselwort wurde verwendet als Name („long“) |
| **short** *riesig***;** | gültig ungültig X | Es ist nicht selbsterklärend (es wird funktionieren, aber es ist nicht praktisch) |
| **boolean** *wahrOderFalsch***;** | gültig X ungültig |  |
| **char** *switch***;** | gültig ungültig X | Ein Schlüsselwort wurde verwendet als Name |
| **long** *ungueltige\_eingaben***;** | gültig X ungültig |  |