

LU01d - Variablen deklarieren

Vorschrift und Wert

$$x = y + z$$

Aus der Algebra kennst du bereits solche Rechnungen mit Platzhaltern. Du weißt, dass jeder Buchstabe stellvertretend für eine bestimmte Zahl steht. Wenn wir diese Gleichung etwas genauer betrachten sehen wir:

- Eine Vorschrift, was zu berechnen ist: $y+z$
- Den Ort, an dem das Ergebnis gespeichert wird: $x=$

Damit du das Ergebnis der Gleichung berechnen kannst, müssen die Platzhalter einen bestimmten Wert erhalten. Wenn du nun wissen, dass $y=4$ und $z=9$ ist, kannst du x berechnen.

Variablen im Programm

In der Programmierung nennen wir die Platzhalter **Variablen**. Variablen reservieren Speicherplatz um darin Werte zu speichern. Wenn du das Ergebnis von $5+7$ speichern willst, musst du eine passende Variable bereitstellen.

In den meisten Programmiersprachen müssen die Variablen **deklariert** werden, bevor sie zum ersten mal verwendet werden.



Die Deklaration legt den Namen (Bezeichner) und Datentyp einer Variable fest. Durch die Deklaration kennt der Compiler die Variable und wir können Sie im Programm verwenden.

Name

Um die Variable zu verwenden, benötigt sie einen Namen (Bezeichner). Der Name einer Variable sollte möglichst sprechend sein. Das bedeutet, dass ihr Name etwas über den Inhalt bzw. Verwendungszweck aussagt. Die Variablennamen in Python sind immer klein geschrieben, nach dem Styleguide von Python. Der Name besteht in der Regel aus Buchstaben, Zahlen und dem Underline (`_`). Variablennamen sollten in `snake_case` geschrieben werden, wobei Wörter durch Unterstriche verbunden werden und alle Buchstaben klein sind.



Gewöhne dich von Anfang an daran, alle Bezeichner in englisch zu schreiben. Verwende niemals Leerzeichen, Umlaute oder Sonderzeichen in einem Variablennamen.

Namenskonventionen am BZZ

Das BZZ hat die folgenden Namenskonventionen:

- Variablen- und Funktionsnamen: Kleinbuchstaben, die durch `_` getrennt sind (`snake_case`).
- Konstanten: Großbuchstaben, die durch `_` getrennt sind (`UPPER_CASE`).

Gute Variablennamen

- `givenname`
- `price`
- `length`
- `student_id` (für eine ID)
- `birth_date` (für ein Datum)

Schlechte Variablennamen

- `i` (Nicht sprechend)
- `länge` (Umlaut)
- `grad°` (Sonderzeichen)
- `Number` (gross geschrieben)
- `PriceInUSD` (CamelCase, sollte in `snake_case` sein)

Übung zu Variablennamen

In einem Programm benötigst du verschiedene Variablen. Lege fest wie die Variablen für folgende Aufgaben heissen sollen:

1. Die Variable speichert die Anzahl der Noten in einem Modul
2. Die Variable speichert eine Schulnote
3. Aus der Anzahl und dem Total der Noten wird der Durchschnitt berechnet. Diese Variable soll den Durchschnitt speichern.
4. Du schreibst ein Programm, dass Lieferscheine erstellt. Dazu benötigst du Variablen um
 - die Bezeichnung eines Artikels zu speichern
 - den Preis eines Artikels zu speichern
 - die Einheit zu speichern, in denen der Artikel verkauft wird (z.B. Stück, Meter, ...)
 - das Total (berechnet aus Preis * Menge) zu speichern
 - speichern zu können, ob es sich beim Artikel um eine Aktion (Preisreduktion) gehandelt hat oder nicht

Datentyp

Der Datentyp bestimmt die Speicherung und spätere Verwendung der Variablen. Je nach Programmiersprache werden unterschiedliche Datentypen unterstützt und die einzelnen Datentypen heissen unterschiedlich. Grundsätzlich lassen sich die Datentypen wie folgt einordnen:

- Textvariablen
 - String aus mehreren Zeichen „Hallo“
- Zahlen
 - Ganzzahlen 5
 - Fließkommazahlen die auch Dezimalbrüche enthalten können 5.53
- Spezielle Datentypen
 - z.B. Ja-Nein (True-False) True

Je nach Programmiersprache wird zusätzlich zwischen einer binären und einer dezimalen Speicherform unterschieden.



Python gehört zu den Programmiersprachen, die den Datentyp selber bestimmen. Daher kannst du auch in Blockly den Datentyp nicht explizit auswählen.

Inhalt bzw. Wert

Jede Variable hat zu jedem Zeitpunkt einen bestimmten Wert. Wenn sie eine Variable vereinbaren (Fachbegriff: deklarieren), so wird im Hauptspeicher ein kleines Stück Speicherplatz für diese Variable reserviert. Wird einer Variablen bei der Deklaration kein Wert zugewiesen, so unterscheidet sich das Verhalten resp. der initiale Wert der Variable je nach Programmiersprache.

Initialisierung

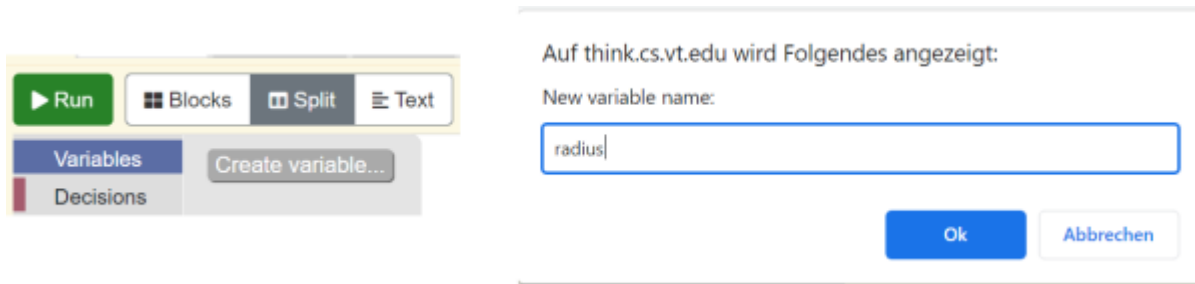
Bei der Initialisierung wird einer Variable direkt bei der Deklaration ein Wert zugewiesen. Dadurch hat die Variable schon beim Start des Programmes einen genau definierten Wert. Dies ist aber nicht in allen Programmiersprachen möglich. In Visual Basic zum Beispiel muss die Initialisierung auf einen Startwert in einer separaten Codezeile durchgeführt werden.

Deklaration

Blockly

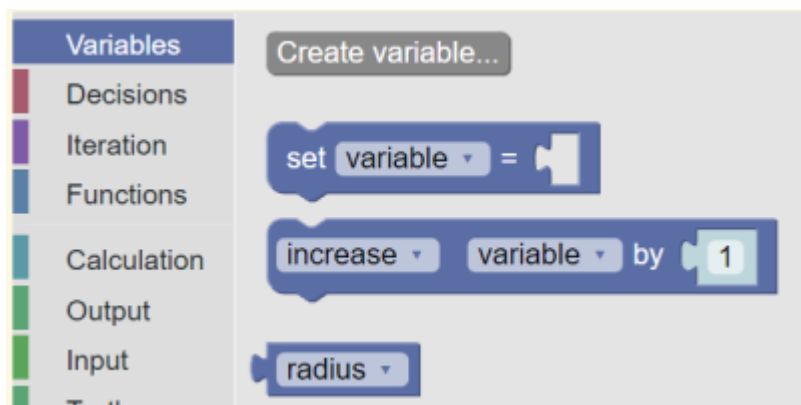
Im Reiter „Variables“ von Blockly kannst du Variablen deklarieren und bestehende Variablen verwenden.

Um eine neue Variable zu deklarieren, wählst du unter „Variables“ den Button [create variable ...]. Dadurch wird ein Popup geöffnet, wo du den Namen der Variable festlegst.



Um eine deklarierte Variable zu verwenden, findest du drei Blöcke:

- `set variable =`: Mit diesem Block kannst du einer Variable einen Wert zuweisen.
- `increase variable by`: Dient zum Rechnen mit Variablen.
- `radius`: Diesen Block brauchst du, um die Variable in einem Befehl einzufügen.



Programmiersprachen

Je nach Programmiersprache ist die Syntax für die Deklaration etwas anders. Nachstehend ein paar Beispiele aus den Sprachen Pascal, Java und Visual Basic:

Pascal

```
var count : integer; // count ist eine Variable um eine Ganzzahl zu speichern
var total : real; // Total ist eine Variable die eine reelle Zahl (Dezimalbrüche) speichert
var givenname : string; // givenname speichert einen Text bzw. Zeichenkette
```

1. Das Schlüsselwort `var` zeigt an, dass eine **Variable** deklariert wird.
2. Der Name oder auch Bezeichner der Variable
3. Nach dem Doppelpunkt folgt der Datentyp

Python

```
count = 1
total = 5.75
```

```
givenname = 'Petra'
```

1. Name der Variable
2. = Wert
3. oder = None

C, Java

```
int count = 1  
double total = 5.75  
String givenname = "Petra"
```

1. Datentyp
2. Name der Variable
3. (optional) = Wert
4. ;

Visual Basic

```
Dim total As Integer
```

1. Dim
2. Name der Variable
3. As
4. Datentyp

Beachte: Je nach Programmiersprache und Ort der Deklaration (Gültigkeitsbereich) **muss** eine Variable zuerst initialisiert werden, bevor sie im weiteren Programmcode z.B. für eine Ausgabe oder eine Bedingung verwendet werden kann.

Übung

Es werden dir einige Variablen mit Bezeichnung, Datentyp und Initialwert vorgegeben. Schreibe für diese Variablen den korrekten Code inklusive Initialisierung in den Programmiersprachen **Python** und **Java**. Verwende dazu das Internet, um die korrekten Notationen zu finden und allenfalls die englische Übersetzung zu finden. Achte auf korrekte Gross-/Kleinschreibung.

1. Vorname (Zeichenkette, „Hans“)
2. Körpergröße (Ganzzahl, 179)
3. Gewicht (Dezimalzahl, 81.5)
4. Verheiratet (Ja-Nein, Ja)
5. Geschlecht als 1-Buchstabe-Kürzel (Zeichen, „m“)

[M319-LU01](#), [M319-C1G](#), [M319-C1F](#)



Marcel Suter, Kevin Maurizi

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
<https://wiki.bzz.ch/de/modul/m319/learningunits/lu01/deklaration>

Last update: **2025/06/23 15:44**

