

LU02c - Einlesen verschiedener Datentypen



Jede Benutzereingabe wird zunächst als Text (String) gelesen. Wollen wir die Eingabe als Zahl oder Wahr/Falsch-Wert verarbeiten, müssen wir den Datentyp konvertieren.

Der Typ der Variable gibt Auskunft über mögliche Werte

Der Typ einer Variablen kann nach dem Lesen von Benutzereingabedaten spezifiziert werden. Eine Variable, die die Zeichenkette „text“ enthält, wird zum Beispiel beim Lesen mit dem input-Aufruf deklariert, und eine Ganzzahl mit dem Wert 42 kann mit `int(variable_name)` deklariert werden.

Der Typ einer Variablen bestimmt die Arten von Werten, die ihr zugewiesen werden können. String-Typen enthalten Text, int-Typen ganze Zahlen, float Gleitkommazahlen, und boolesche Typen sind entweder wahr oder falsch.

Die möglichen Werte eines bestimmten Variablentyps sind also begrenzt. Eine String-Variable kann z. B. keine Integerwerte enthalten, und eine Float-Variable kann keinen booleschen Wert enthalten.

Typ	Beispiel	Akzeptierte Werte
Ganze Zahl, d.h. <code>int</code>	<code>value = 4</code>	Eine Integer-Zahl kann jede ganze Zahl enthalten. Einige Programmiersprachen haben Grenzen für Integer-Werte, die mit 32- oder 64-Bit-Darstellungen zu tun haben, aber seit Python 3 dürfen die Werte von <code>int</code> unbeschränkt sein.
Fließkommazahl, d.h. <code>float</code>	<code>value = 4.2</code>	Fließkommazahlen enthalten Dezimalzahlen, wobei der größtmögliche Wert $1,7976931348623157e+308$ (ein riesiger Wert) ist. Wenn eine Dezimalzahl durch eine Fließkommazahl dargestellt wird, kann der Wert ungenau sein, da Fließkommazahlen nicht in der Lage sind, alle Dezimalzahlen darzustellen. Sie können diese Ungenauigkeit sehen, wenn Sie (zum Beispiel) den Befehl <code>1.2 - 1.0</code> in einer Python-Datei ausführen. Solange Sie keine ernsthaften wissenschaftlichen Berechnungen durchführen, sollten Sie sich in der Praxis keine großen Sorgen über diese Fehler machen.
String	<code>value = 'Hi'</code>	Eine Zeichenkette kann Text enthalten. Zeichenketten werden in Anführungszeichen gesetzt.
Wahrer oder falscher Wert, d.h. <code>bool</code>	<code>right = True</code>	Ein Boolean enthält entweder den Wert <code>True</code> oder <code>False</code> . (Beachten Sie die Großschreibung.)

Zeichenkette lesen

Das Lesen einer Zeichenkette ist der einfachste Fall, da jede Benutzereingabe zunächst sowieso als Zeichenkette gelesen wird.

```
set animal = input ( " Tierart? " )
```

Ganzzahlen lesen

Der Befehl `int` wandelt eine Zeichenkette in eine ganze Zahl um. Er nimmt die Zeichenkette, die den zu konvertierenden Wert enthält, als Parameter.

```
set input_string = input ( " Alter? " )  
set age = int ( input_string )
```

In diesem Beispiel wird die Benutzereingabe zunächst in eine String-Variable `input_string` gelesen. Anschliessend wird der Wert von `input_string` mit der Funktion `int(...)` in eine Ganzzahl konvertiert und in `age` gespeichert.

```
input_string = input('Alter?')  
age = int(input_string)
```

Du kannst die Benutzereingabe und die Konvertierung auch in einem Befehl kombinieren:

```
set age = int ( input ( " Alter? " ) )
```

```
age = int(input('Alter?'))
```

Flieskommazahlen und Boolesche Werte lesen

Die Befehle `float` und `bool` konvertieren Zeichenketten in Gleitkommazahlen und Boolesche Werte. Ähnlich wie `int` nehmen sie die Zeichenkette, die den zu konvertierenden Wert enthält, als Parameter.

```
set length = float ( input ( " Länge? " ) )
```

```
set exists = bool ( input ( " Existiert? " ) )
```

```
length = float(input('Länge?'))  
exists = bool(input('Existiert?'))
```

```
value_as_string = "True"  
value = bool(value_as_string)
```

Fehler beim Konvertieren

Damit Python einen Wert nach `int`, `float` oder `bool` konvertieren kann, muss dieser natürlich gültig sein. Gibt der Benutzer zum Beispiel als Länge „viel“ ein, wird die Konvertierung mit einer Fehlermeldung scheitern.

[M319-LU02](#), [M319-B1G](#), [M319-B1F](#), [M319-C1G](#), [M319-C1F](#)



© Kevin Maurizi, Marcel Suter

Diese Theorieseite basiert auf einer übersetzten Seite von [Scott Morgan](#), verwendet unter CC BY NC SA.

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

Permanent link:
<https://wiki.bzz.ch/modul/m319/learningunits/lu02/verschiedenedatentypen>

Last update: **2024/03/28 14:07**

