

# Python

## Dateiverarbeitung

Lesen	<pre>with open("file.txt", "r") as datei:     text = datei.read()</pre>
Schreiben	<pre>with open("file.txt", "w") as datei:     datei.write(text + "Hallo\n")</pre>
Leerzeichen entfernen	<pre>text.strip("\n")</pre>

## Nutzerinteraktion

<pre>input() input("str")</pre>	<p>Unterbricht den Programmablauf und wartet auf Eingabe des Users. Programm wird danach fortgesetzt. Der Funktion kann ein String übergeben werden, der auf dem Bildschirm angezeigt wird. Die Eingabe des Users ist der Rückgabewert dieser Funktion. Rückgabewert ist immer String.</p>	<pre># Eingabeaufforderung # Eingabe des Users wird in # abfrage geschrieben abfrage = input("Hier könnte Ihre Werbung stehen: ")  # Bildschirmausgabe Hier könnte Ihre Werbung stehen: # Usereingabe Nein  print(abfrage) # Ausgabe Nein</pre>
-------------------------------------	--	---

## Verzweigungen

```

if bmi < 19:
    print("Untergewicht")
elif bmi <= 24:
    print("Normalgewicht")
else:
    print("Uebergewicht")

```

## Schleifen

while	<pre> i = 0 while i &lt; 10:     print(i)     i += 1 </pre>
for (eigentlich foreach)	<pre> for zahl in [1, 2, 3]:     print(zahl) </pre>

## Operatoren

in	<p>Prüft ob Variable in angegebener Sequenz vorhanden ist          vorhanden gibt "True" zurück          nicht vorhanden gibt "False" zurück</p>	<pre> #true print(2 in [2, 3, 4])  #false erg = "bla" in ["bli", "blub"] print(erg) </pre>
range	<p>Funktion gibt eine Sequenz von Zahlen zurück  <code>range(start, stop, step)</code></p> <p>start = inklusiver Anfang der Sequenz, optional, default ist 0          stop = exklusives Ende der Sequenz, benötigt          step = Inkrement, optional, default 1</p>	<pre> # 0, 1, 2, 3, 4 range(5)  # 4, 5, 6, 7 range(4, 8)  # 2, 4, 6, 8 range(2, 9, 2) </pre>

## Listen

append()	hängt Objekt an das Ende der Liste	<pre># hängt den Wert 5 an Liste an liste.append(5)</pre>
index()	gibt den ersten Index zu angegebenem Wert zurück	<pre>liste = [2, 3, 4, 5, 6, 3, 4, 5] print(liste.index(4)) # Ausgabe ist 2</pre>
insert()	fügt Wert vor Index ein <code>liste.insert(index, value)</code>	<pre>liste = [2, 3, 4, 5] liste.insert(1, 6) print(liste) # Ausgabe [2, 6, 3, 4, 5]</pre>
remove()	entfernt das erste Vorkommen des Werts	<pre>liste = [2, 3, 4, 5] liste.remove(4) print(liste) # Ausgabe [2, 3, 5]</pre>
reverse()	Umdrehen der Liste in-place	<pre>liste = [2, 3, 4, 5] liste.reverse() print(liste) # Ausgabe [5, 4, 3, 2]</pre>

<p>sort()</p>	<p>Sortiert die Liste in-place  default = aufsteigend  "reverse=True" = absteigend</p>	<pre># Sortieren in aufsteigender Reihenfolge liste = [3, 2, 5, 4] liste.sort() print(liste) # Ausgabe [2, 3, 4, 5]  # Sortieren in absteigender Reihenfolge liste = [3, 2, 5, 4] liste.sort(reverse=True) print(liste) # Ausgabe [5, 4, 3, 2]</pre>
<p>Ausgabe Formatierung</p>	<p>Die Ausgabe von Listen kann nach folgender Notation formatiert werden  <code>listenname[start:stop:step]</code></p> <p>start = inklusiver Startwert, default 0  stop = exklusiver Stopwert, default Länge  step = Inkrement, default 1</p>	<pre>liste = [2, 3, 4, 5, 6, 7, 8, 9]  print(liste[0:4]) # Ausgabe [2, 3, 4]  print(liste[1:6:2]) # Ausgabe [3, 5, 7]  print(liste[2:6:2]) # Ausgabe [4, 6]</pre>

## Umwandlung von Datentypen

Wichtig: Alle Funktionen sind **KEINE** in-place Ersetzung

type()	Gibt den Datentyp einer Variablen zurück	<pre>type(5) # Ausgabe &lt;class 'int'&gt;  type(5.0) # Ausgabe &lt;class 'float'&gt;  type(5) # Ausgabe &lt;class 'str'&gt;</pre>
str()	Rückgabewert dieser Funktion ist ein String	<pre>str(5) # Ausgabe '5'</pre>
int()	Rückgabewert dieser Funktion ist ein Integer	<pre>int("5") # Ausgabe 5</pre>
float()	Rückgabewert dieser Funktion ist eine Fließkommazahl	<pre>float(5) # Ausgabe 5.0</pre>

---

Revision #16

Created 2022-08-01 08:56:26 UTC

Updated 2022-08-02 08:41:07 UTC