

Woche 03: Programmierung – Einführung in Listen

Skript

Erarbeitet von
Ludmila Himmelpach

Lernziele	1
Inhalt	1
Einstieg.....	1
Eigenschaften von Listen	2
Aufbau von Listen	2
Verschachtelte Listen.....	3
Take-Home Message	4
Weiterführendes Material.....	4
Disclaimer	4

Lernziele

- Definieren können, was eine Liste ist
- Erklären können, wie eine Liste in Python aufgebaut ist

Inhalt

Einstieg

Beim Umzug werden Sachen, die zusammen gehören, meistens in eine Kiste gepackt. So kommen zum Beispiel Bücher in eine Kiste und Geschirr in eine andere Kiste. Mit den Daten in einem Programm funktioniert das leider nicht so einfach, weil pro Variable nur ein Datenobjekt gespeichert werden kann.

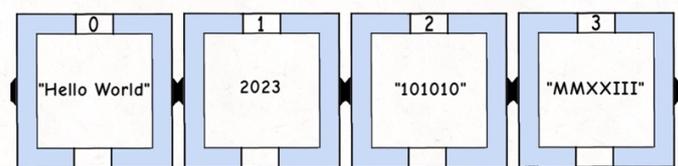
Wenn wir uns eine Variable als eine Kiste vorstellen, in der Datenobjekte im Arbeitsspeicher eines Computers aufbewahrt werden, können wir in eine solche Kiste nur ein Datenobjekt, zum Beispiel ein Buch bzw. einen Buchtext, reinlegen.

Wenn wir mehrere Bücher ablegen wollen, brauchen wir also für jedes Buch eine eigene Kiste. Das wird einerseits mit der steigenden Anzahl der Datenobjekte schnell unübersichtlich und andererseits besteht kein Zusammenhang zwischen den Variablen, die zusammenhängende Daten enthalten. Außerdem müsste man sich alle Variablennamen bzw. Kistenbeschriftungen merken, um auf deren Inhalte später zugreifen zu können.

Um etwas Ordnung zu schaffen, würde man wahrscheinlich eine Liste mit allen Variablennamen erstellen und die Verknüpfungen zwischen ihnen selbst eintragen. Das braucht man aber nicht, weil Python einige Datenstrukturen selbst anbietet, die zur Organisation und Speicherung der Daten dienen. Die wahrscheinlich am meisten benutzte Datenstruktur heißt *Liste*.

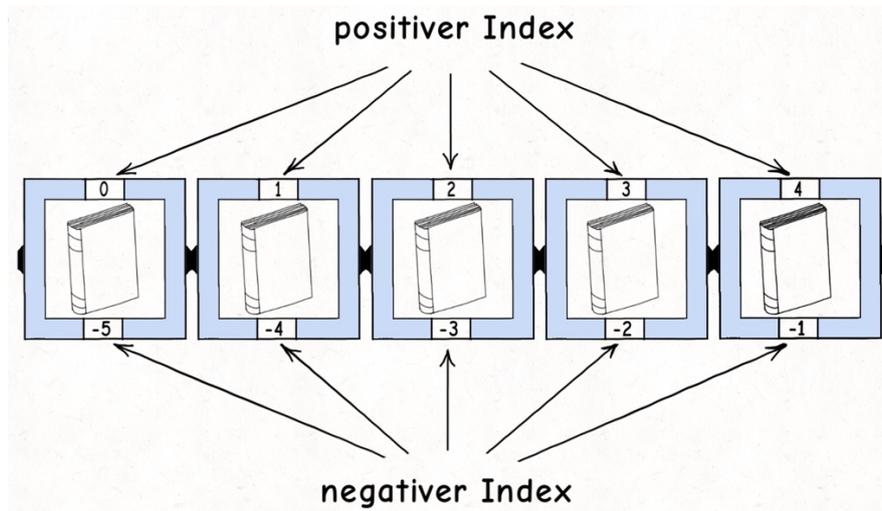
Eigenschaften von Listen

Listen beinhalten getrennte Datenobjekte, die an einem Ort gespeichert werden. Damit man die Listen später wiederfinden kann, bekommen sie wie Variablen beim Anlegen einen Namen. Listen sind in Python sehr flexibel. In einer Liste können Datenobjekte unterschiedlichen Typs, wie zum Beispiel Zeichenketten oder Zahlen, gespeichert werden. Listen können jederzeit erweitert oder auch gekürzt werden, indem Datenobjekte oder auch ganze Listen einer Liste hinzugefügt oder aus dieser entfernt werden. In einer Liste können auch Datenobjekte durch andere Datenobjekte an beliebiger Stelle ersetzt werden.

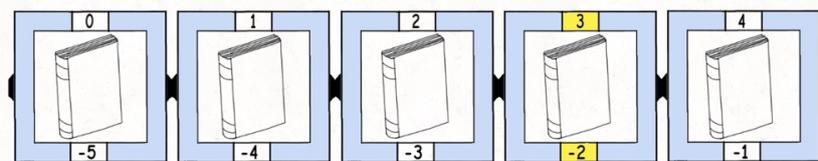


Aufbau von Listen

Damit man auf die Datenobjekte innerhalb einer Liste direkt zugreifen kann, wird jedem Objekt ein Index zugewiesen, der seine Position bzw. Adresse in der Liste bestimmt. Genaugenommen besitzt jedes Datenobjekt zwei Indizes innerhalb einer Liste, einen positiven und einen negativen.



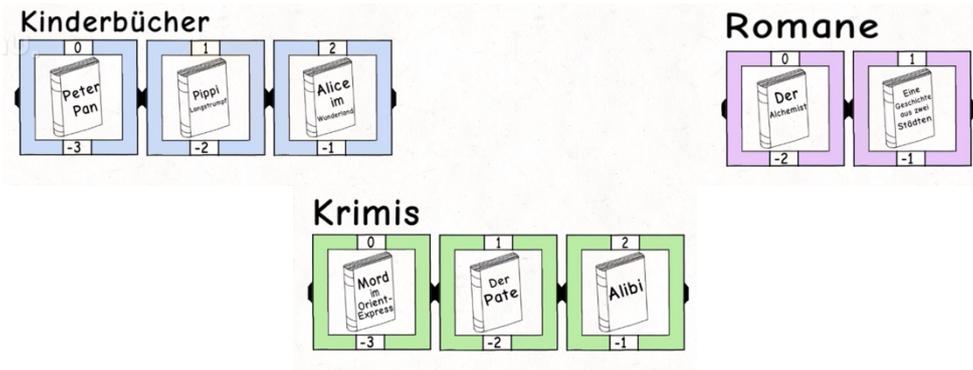
Der Index wird als positiver Index interpretiert, wenn dieser von null beginnend die Liste von links nach rechts durchläuft. Wird die Liste von rechts nach links durchlaufen, spricht man von einem negativen Index, der auf der rechten Seite mit -1 anfängt. Also kann jedes Datenelement innerhalb einer Liste auf mindestens zwei Weisen adressiert werden. In dieser Liste mit fünf Büchern findet man das erste Buch entweder unter der Adresse 0 oder -5. Das vorletzte Buch hat 3 als positiven Index und -2 als negativen Index.



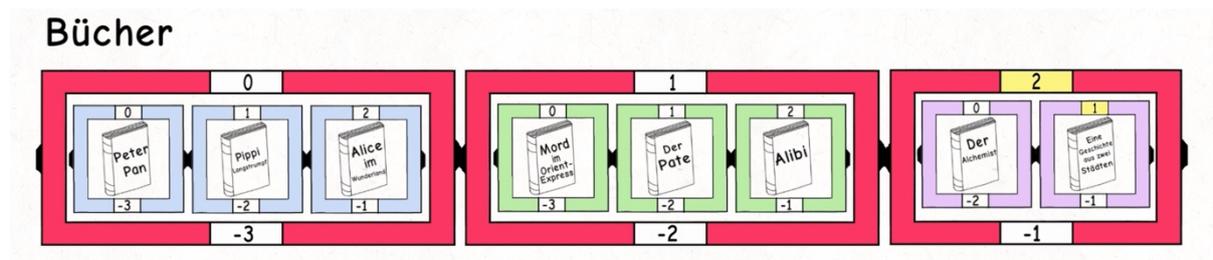
Verschachtelte Listen

In Python können Listen nicht nur einzelne Datenobjekte, sondern auch ganze Listen als Elemente aufnehmen. Auf diese Weise entstehen Listen von Listen oder auch Listen von Listen von Listen usw., wenn man mehr Hierarchieebenen braucht. Da jedes Element einer Liste über seinen Index gefunden werden kann, bekommt jedes Element einer aufgenommenen Unterliste eine weitere Indextebene. Bei einer Liste, die selbst aus Listen von Datenobjekten besteht, müssen also zwei Indextzahlen angegeben werden, um auf Datenobjekte zugreifen zu können. Das sieht man am besten an einem Beispiel.

Nehmen wir an, wir haben ganz viele Bücher, die bereits nach Themen in drei Listen zusammengefasst sind. Wir haben eine Liste mit Kinderbüchern („Peter Pan“, „Pippi Langstrumpf“, „Alice im Wunderland“), eine weitere Liste mit Krimis („Mord im Orient-Express“, „Der Pate“, „Alibi“, „Der dünne Mann“) und die dritte Liste mit Romanen („Der Alchemist“, „Eine Geschichte aus zwei Städten“). Jedes Buch kann durch einen Index in seiner Liste angesprochen werden. Zum Beispiel „Pippi Langstrumpf“ findet man unter Index 1 oder -2 in der Liste der Kinderbücher. Das Buch „Alibi“ kann entweder durch seinen positiven Index 2 oder den negativen Index -1 in der Krimi-Liste angesprochen werden.



Wenn wir diese drei Listen als Elemente einer allgemeinen Bücherliste aufnehmen, bekommen die Unterlisten selbst Indizes in der übergeordneten Liste. Um jetzt ein bestimmtes Buch abrufen zu können, muss man zwei Indexzahlen angeben. Die erste Zahl gibt den Index der Unterliste an, die zweite entspricht der Position des Elements in der Unterliste. Der Roman „Eine Geschichte aus zwei Städten“ ist jetzt unter dem Index 2.1 zu finden, wobei die 2 die Position der Liste mit Romanen in der Bücherliste angibt und die 1 die Position des Buches innerhalb seiner Liste.



Take-Home Message

In diesem Video hast Du den allgemeinen Aufbau einer Liste kennengelernt. Im nächsten Video lernst Du, wie man eine Liste in Python erzeugt und welche Operationen auf Listen möglich sind.

Weiterführendes Material

Schmitt, S. (2021). *Python Kompendium: Professionell Python Programmieren lernen*. BMU Media Verlag

Barry, P. (2017). *Python von Kopf bis Fuß*. O'Reilly

Disclaimer

Transkript zu dem Video „Woche 03: Programmierung – Einführung in Listen“, Ludmila Himmelpach.

Dieses Transkript wurde im Rahmen des Projekts ai4all des Heine Center for Artificial Intelligence and Data Science (HeiCAD) an der Heinrich-Heine-Universität Düsseldorf unter der Creative Commons Lizenz [CC-BY](https://creativecommons.org/licenses/by/4.0/) 4.0 veröffentlicht. Ausgenommen von der Lizenz sind die verwendeten Logos, alle in den Quellen ausgewiesenen Fremdmaterialien sowie alle als Quellen gekennzeichneten Elemente.