

# Linked Stack

Einen Stapel kann man folgendermaßen modellieren:



Das sieht also ganz ähnlich aus, wie bei der verketteten Liste. Auch der Stapel/Stack arbeitet mit einer Struktur von Knoten, die „aufeinander zeigen“, er kann also wie die Liste beliebig viele Objekte verwalten. Der Stapel muss sich jedoch in seinem Attribut head nur eine Referenz merken - die auf den **obersten Knoten** des Stapels. Alle Operationen des Stacks finden immer auf diesem obersten Knoten statt:



## Implementation

### Kickoff

Zu implementieren sind also die Operationen push, pop, top und isEmpty. Hier: <https://codeberg.org/qg-info-unterricht/bluej-linked-stack> findest du eine BlueJ-Vorlage, in der du arbeiten kannst.

- Bearbeite die Klasse LinkedStack
- Verwende wie schon bei der verketteten Liste die Knotenklasse zur Datenspeicherung.
- Du musst zunächst Attribut(e) und Konstruktor des LinkedStack implementieren, bevor du mit den Methoden des Stacks beginnen kannst - bedenke folgendes:
  - head muss stets auf den obersten Node des Stacks zeigen
  - Die Nodes sind vom Typ Node<T> (schau nach was der Diamantoperator macht, wenn du das nicht mehr weißt).
  - Der Konstruktor muss einen leeren Stack erzeugen - auf was zeigt dann head?



### (A1) Konstruktor und Attribute

Erweitere die Klasse so, dass es sinnvoll möglich ist, ein neues Stack Objekt zu erzeugen.

### push - Element auf den Stapel legen

Das folgende Schaubild veranschaulicht den Vorgang, wenn ein neues Element auf den Stapel gelegt werden soll.



# Material

[n/a: Keine Treffer]

From:

<https://wiki.qg-moessingen.de/> - **QG Wiki**

Permanent link:

<https://wiki.qg-moessingen.de/faecher:informatik:oberstufe:adt:stack:linkedstack:start?rev=1633546673>

Last update: **06.10.2021 20:57**

