

# Datenstrukturen

#SEW

#GEI

Datenstrukturen werden benötigt um Daten zu speichern, zu organisieren und zu verwalten. Sie ermöglichen die effiziente Verarbeitung von Daten und sind grundlegend für die effektive Implementierung von Algorithmen.

Bsp: Suchen, Sortieren...

**Die wichtigsten Vertreter sind:**

List (z.B. ArrayList, LinkedList...)

- Doppelte Objekte erlaubt (können abgespeichert werden), Reihenfolge bleibt erhalten, komfortable Alternative zu Arrays

```
List <String> stringList = new ArrayList<>();
stringList.add("Bruh");
```

Set (z.B. HashSet, TreeSet...)

- Menge mit Objekten in der jedes Element nur einmal vorkommt

```
Set <String> nameSet = new Set<>();
nameSet.add("Baracko bama");
```

Map (z.B. HashMap)

- Helfen bei der Speicherung von Schlüssel-Wert Paaren (Key-Value-Pairs)

```
Map <String, String> capitalCities = new HashMap<String, String>();

// add keys and values
capitalCities.put("Austria", "Vienna");
```

Queue

- Realisierung von FIFO (first-in first-out) und LIFO (last-in first-out) Speicher

## Datenstrukturen Konzepte

Konzept		Interface		
	List	Queue	Set	Map
Arrays	ArrayList			
verkettete Listen	LinkedList	LinkedList		
Bäume			TreeSet	TreeMap
Hashtabellen			HashSet	HashMap

## Verkettete Liste

Eine Verkettete Liste ist eine dynamische Datenstruktur in der Datenelemente geordnet gespeichert sind.



## Binäre Bäume

