

# List Set oder Map?

Wann verwende ich was?

# List

- **Allgemein:**
- Anschaulich: Es ist eine Liste -> Einkaufsliste
- Doppelte Elemente erlaubt
- Elemente haben einen Index -> get(Index)
- Objekte haben eine feste Reihenfolge
- **Eigenschaften:**
- Ein Objekt nach einem Index suchen, ist sehr schnell
- Ein Objekt ohne Index suchen ist langsam
- Iteration durch eine Liste ist relativ schnell
- Man kann eine Liste sortieren
- **Klassen:**
- **ArrayList** -> Wenn man eine Liste aufbauen will und nur Elemente hinzufügt -> sprich man will Daten ablegen aber diese kaum verändern
- **LinkedList** -> Wenn man eine Liste aufbauen will und dort ständig Elemente hinzufügt oder löscht

# Set

- **Allgemein:**
  - Stell dir das Set vor wie eine Mathematisch Menge -> Ein Kreis wo die Elemente hineinkommen
  - Jedes Element ist einzigartig d.h. es kommt nur einmal darin vor
  - Kein Index wie List
  - Sehr schnell falls man nach einem bestimmten Element sucht
  - Eigene Objekte -> hashCode() und equals() muss implementiert werden
- **Klassen:**
  - **HashSet** -> Nutzt das Hash Verfahren, Nutzt man wenn die Reihenfolge egal ist
  - **TreeSet** -> Sortiert in der natürlichen Reihenfolge (1,2,3 oder a,b,c)
  - **LinkedHashSet** -> Ordnet die Elemente so an wie diese hinzugefügt wurden

# Map

- **Allgemein:**
- Jedes Element erhält einen eigene Schlüssel
- Es gibt keine doppelten Schlüssel
- Abrufen eines Elementes mittels des Schlüssels ist sehr schnell
- Die Iteration über eine Map dagegen ist langsam, genau genommen über die Werte einer Map
- Wenn Du über eine Map iterieren musst, dann immer über die Schlüssel / Keys -> keySet()
- Wenn Du eigene Objekte nutzen möchtest musst die hashCode() und equals() implementieren
- **Klassen:**
- **HashMap** -> Nutzt das Hash Verfahren, Nutzt man wenn die Reihenfolge egal ist
- **TreeMap** -> Sortiert in der natürlichen Reihenfolge (1,2,3 oder a,b,c)
- **LinkedHashMap** -> Ordnet die Elemente so an wie diese hinzugefügt wurden

# Beispiele:

Wann verwende ich z.B.

## **List:**

To-Do Liste, Kontakte speichern, Einkaufsliste,

## **Set:**

In der App Entwicklung z.B. um die Daten abzulegen wo der User gedrückt hat. Um Abfragen zu starten hat der wirklich da drauf gedrückt -> dann passiert das...

## **Map:**

Endgegner speichern, jeder Gegner hat seinen Key über diese Key lade ich den Gegner in das jeweilige Level.  
Jeder Artikel/gruppe hat sein/e eigene Artikelnummer