

# Aufgabe 1

Gegeben ist eine Liste mit Wünschen:

```
List<String> wuensche = List.of(
    "Schlitten",
    "Puppe",
    "Schokolade",
    "Snowboard",
    "Playmobil"
);
```

Aufgabe:

Erzeuge einen Stream über die Liste.

Filtere alle Wünsche, die mit „S“ beginnen.

Wandle sie in Großbuchstaben um.

Gib sie mit `forEach` aus.

Beispielausgabe:

```
SCHLITTEN
SCHOKOLADE
SNOWBOARD
```

## Aufgabe 2 – Streams über Objekte

Gegeben ist die Klasse `Geschenk` (siehe Git) und folgende Liste:

```
List<Geschenk> band = List.of(
    new Geschenk("Buch", 500),
    new Geschenk("Auto-Spielzeug", 300),
    new Geschenk("Schokolade", 120),
    new Geschenk("Lego-Set", 900)
);
```

Aufgabe:

1. Filtere alle Geschenke, die **unter 500g** wiegen.
2. Sortiere die gefilterten Geschenke nach Gewicht.
3. Gib sie aus.

## Aufgabe 3 – Streams mit verschachtelten Objekten

Gegeben ist die Klasse Palette (siehe Git) und folgende Liste:

```
List<Palette> lager = List.of(  
    new Palette("A1", new Geschenk("Roboter", 800)),  
    new Palette("A2", new Geschenk("Schokolade", 120)),  
    new Palette("B1", new Geschenk("Rodel", 1100)),  
    new Palette("C3", new Geschenk("Puzzle", 300))  
);
```

**Aufgabe:**

1. Filtere alle Geschenke, die **unter 500g** wiegen.
2. Sortiere die gefilterten Geschenke nach Gewicht.
3. Gib sie aus.

## Aufgabe 4 – collect() anwenden

Nutze dieselbe `lager`-Liste aus Aufgabe 4.

**Aufgabe:**

Erzeuge mittels Streams eine `List<String>`, die nur die **Namen der Geschenke unter 400g** enthält.

**Beispielausgabe:**

```
[Schokolade, Puzzle]
```

## Aufgabe 5 – Imperatives Äquivalent

Gegeben ist:

```
var arr = new String[]{"Chris", "Alex", "Mike"};  
  
Arrays.stream(arr)  
    .map(item -> item.toLowerCase())  
    .filter(item -> item.contains("i"))  
    .map(item -> item + " " + item.length())  
    .forEach(item -> System.out.println(item));
```

**Aufgabe:**

Schreiben Sie ein **imperatives Äquivalent**, ohne Streams, also nur:

- `for`-Schleifen
- `if`-Abfragen

- temporäre Listen oder Arrays

**Hinweis:**

Starte mit: