

## Programmieren 2

### Übung 4

#### Ziel der Übung

- > Multithreading
- > Exception Handling
- > Streams

#### Aufgabenstellung

Sie finden ein Grundgerüst für die Aufgabe unter:

<https://github.com/leonardo1710/NewsreaderStarterExercise4>

Diese Aufgabe baut auf die Übung 3 auf. Das **newsapi** stellt URLs der Nachrichtenquelle und URLs zu Bildern zur Verfügung. Das Ziel dieser Übung ist es die Seiten zu laden und abzuspeichern. Da es bis zu 100 Seiten sein können ist das Ziel der Übung den Download der Seiten (und Bilder) zu parallelisieren und dadurch zu beschleunigen.

Für die Übung werden zwei Klassen zur Verfügung gestellt:

1. **Downloader**: Gibt die Struktur eines Downloaders vor und implementiert die Methode zum Speichern der geladenen Daten.
2. **SequentialDownloader**: Speichert übergebene URLs sequenziell ab.

#### Folgende Punkte sind in der Übung durchzuführen:

- Erweitern Sie das Menü um die Optionen „Download last search“.
- Der **Downloader** ist als Strategy Pattern umgesetzt, wobei der **SequentialDownloader** bereits implementiert ist.
- Schreiben Sie eine Methode, die alle URLs in einer Liste zusammenfasst (Streams).
- Legen Sie ein Verzeichnis „download“ im Projekt an (Wichtig! Auch zum Ausführen des **SequentialDownloader**).
- Die Methode im Controller soll unabhängig vom Typ des Downloader arbeiten.
- Erstellen Sie die Klasse **ParallelDownloader** und implementieren Sie die notwendigen Methoden. Verwenden Sie folgende Java Konzepte:
  - **ExecutorService**: Dieses Interface ermöglicht einen Threadpool anzulegen
  - **Future**: Dieses Interface erlaubt Variablen für laufende Threads zu definieren und trotzdem im Ablauf des Programms fortzufahren.
- Messen Sie die Zeiten für den sequenziellen und parallelen Ablauf und vergleichen diese.

#### Erweiterung

- > Verbesserung des Exception Handling

#### Hinweise

- > Beispiel-Implementierungen für Multithreading finden Sie hier:  
<https://github.com/leonardo1710/ThreadsExample>
- > Weitere Ressourcen zum Thema Thread Pooling inkl. Beispielen:
  - > <https://www.scientecheasy.com/2020/08/java-thread-pool.html/>

> <https://www.baeldung.com/thread-pool-java-and-guava>

## Abgabe

- Exportieren Sie Ihr Git-Repository mit allen Klassen und laden Sie es auf Moodle hoch (.zip oder .rar)
- Geben Sie einen Link zu Ihrem Repository in einem readme.md an
- Das Repository muss public sein!