



JSON visualisieren mit dem json.tool

1 Vorbereitung

Die Standardbibliothek von Python bringt ein Kommandozeilentool mit, das den Inhalt von JSON-Dateien visualisiert bzw. überprüft, ob valides JSON vorliegt. Die Hilfe des Tools ruft man auf mit

```
python3 -m json.tool -h
```

bzw. mit

```
python3 -m json.tool --help
```

2 Übungen

2.1 Übung 1

Lassen Sie sich den Inhalt der Datei **scify_movies.json** auf der Kommandozeile ausgeben. Was bewirkt die Option `--no-ensure-ascii` des `json.tool`?

2.2 Übung 2

Der Inhalt der Datei **order_1line.json** ist ein JSON-Object, das in einer einzigen Zeile steht. Lassen Sie diese JSON-Datei in einer lesbaren „Pretty-Print“-Ansicht formatieren und schreiben sie die Ausgabe in eine neue Datei mit dem Namen **order.json**.

2.3 Übung 3

Das Format ndjson (New Line Delimited JSON) erlaubt pro Zeile ein weiteres JSON-Object und wird z.B. für Logfiles verwendet. Ein Beispiel ist die Elasticsearch Bulk API (<https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-bulk.html>). Verwenden Sie das `json.tool` mit der passenden Option, um die Datei **elasticsearch_bulk_example.ndjson** auszugeben.

2.4 Übung 4

Versuchen Sie die Dateien **order_example_2.json** und **order_example_3.json** zu öffnen. Was sagt Ihnen die Rückmeldung des `json.tool`?