
MT AG
data . cloud . mobile
JSON in Java mit
Schema und JsonPath
Wolfgang Nast

Agenda

- JSON Schema
- Java und
  - JSON Schema
  - JSON Path
  - Mapping
- Zusammenfassung

Im Überblick

Technologie-orientiert
Branchen-unabhängig

Hauptsitz
Ratingen

240
Beschäftigte

MT AG

Zertifizierter
Partner von
Oracle,
Microsoft
und SAP

Inhabergeführte
Aktiengesellschaft

Gründungsjahr
1994

Ausbildungs-
betrieb

kununu
TOP COMPANY
www.kununu.com
VON MITARBEITERN EMPFOHLEN!

Niederlassung
Frankfurt am Main

MT AG

JSON Schema

- JSON Schema Implementierung in Java
- JSON Schema
- JSON lesen und validieren

JSON Schema

- JSON Schema Versionen
- JSON Schema Beispiel

JSON Schema

Vorgaben von Java

- JSON Schema lesen:
  - Java Implementierungen
  - Draft 4
  - Draft 6

JSON Schema

Beschreibung JSON

- JSON definiert die Strukturen:
  - Objekte
  - Arrays
  - Werte
    - String
    - Number
    - Boolean
    - Null

JSON Schema

Beschreibung JSON Schema Strukturen

- JSON Schema definiert die Strukturen und Formate:
  - Number
    - Integer
  - String
    - Enum
    - Date-Time, Date, Time
    - Email
    - …

Beispiel: JSON Schema

JSON Schema Buch

- Teile des Schemas:
  - Allgemeiner Teil
  - Kapitel
  - Autor
  - Buch

Allgemeiner Teil Header

JSON Schema Header

```
{
  "$schema": "http://json-schema.org/draft-06/schema#",
  "title": "Buch",
  "description": "Ein Buch mit Autor und Kapiteln",
  "type" : "object",
  "properties" : {
    "buch" : {"$ref" : "#/definitions/buch" }
  },
```

Allgemeiner Teil Definitionen

JSON Schema Definitionen Id

```
"definitions" : {
   "id" : {
     "id" : "integer",
     "minimum" : 1
   },
```


MT AG

Kapitel Teil 1

JSON Schema

```
"kapitel" : {
  "type": "object",
  "properties": {
    "id": {
      "$ref": "#/definitions/id"
    },
    "name": {
      "type" : "string"
    },
```


MT AG

Kapitel Teil 2

JSON Schema

```
        "text": {
          "type" : "string"
        }
      },
      "required": ["id", "name"],
      "additionalProperties": false
    },
```


MT AG

Autor Teil 1

JSON Schema

```
"autor" : {
  "type" : "object",
  "properties" : {
    "id" : {"$ref" : "#/definitions/id"},
    "name" : {
      "type" : "string"
    },
    "geburtstag" : {
      "type" : "string",
      "format" : "date"
    },
```


MT AG

Autor Teil 2

JSON Schema

```
        "geschlecht" : {
          "enum" : ["frau", "mann"]
        }
      },
      "required": ["id", "name", "geschlecht"],
      "additionalProperties": false
    },
```


MT AG

Buch Teil 1

JSON Schema

```
"buch" : {
  "type" : "object",
  "properties" : {
    "id": {
      "$ref": "#/definitions/id"
    },
    "name" : {
      "type" : "string"
    },
```


MT AG

Buch Teil 2

JSON Schema

```
      "kalitel" : {
        "type" : "array",
        "items": {
          "$ref": "#/definitions/kapitel"
        },
        "minItems": 1,
        "uniqueItems": true
      }
    },
```


MT AG

Buch Teil 3

JSON Schema

```
        "required": ["id", "name"],
        "additionalProperties": false
      }
    }
}
```


MT AG

Java und JSON Schema

- Implementierungen auf GitHub
  - everit-org / json-schema
  - java-json-tools / json-schema-validator
  - networknt / json-schema-validator

JSON Schema und everit-org

Schema Validierung

- everit-org
  - Draft 4 und Draft 6
  - json.org
  - Format(date-time, email,…)

JSON Schema und java-json-tools

Schema Validierung

- java-json-tools
  - Draft 4
  - Jackson Node
  - Kein Format

JSON Schema und networknt

Schema Validierung

- networknt
  - Draft 4
  - Jackson Node
  - Kein Format
  - Schnelle Validierung

JSON Path

- Operatoren
- Funktionen
- Filter Operatoren

JSON Path Operatoren Teil 1

Operator	Beschreibung
$	Root Element (immer anzugeben)
@	Aktueller Node
*	Wildcard (Name oder Nummer)
..	Deep Scan
.<Name>	Unterelement mit Name
['<Name1>'(, '<Name2>')]	Unterelemente mit mehreren Namen

JSON Path Operatoren Teil 2

Operator	Beschreibung
[<Nummer1>(,<Nummer2>)]	Index im Array (auch mehrere)
[<von>:<bis>]	Index Block im Array(von-bis)
[?(<Filter>)]	Filteranweisung

JSON Path Funktionen

Funktionen für Array Variablen

Funktionen	Beschreibung
min()	Minimaler Wert im Array
max()	Maximaler Wert im Array
avg()	Mitterwert von der Werten des Arrays
stddev()	Standarabweichung der Werte
length()	Länge des Arrays

JSON Path Filter Operatoren Teil 1

Operatoren für den Filter (?)

Funktionen	Beschreibung
==	Links und Rechts sind gleich
!=	Links und Rechts unterscheiden sich
<	Links ist kleiner als Rechts
<=	Links ist kleinergleich zu Rechts
>	Links ist größer als Rechts
>=	Links ist größergleich zu Rechts
=~	Links entspricht den Regulären Ausdruck

MT AG

JSON Path Filter Operatoren Teil 2

Operatoren für den Filter (?)

Funktionen	Beschreibung
in	Links ist Teil der Liste
nin	Links ist nicht Teil der Liste
subsetof	Links ist Teilliste der Liste
size	Links ist Array der Länge oder String der Länge
empty	Links ist leeres Array oder ein Leerstring

JSON Path Beispiele

JSON Path	Beschreibung
$.kapitel[0].name	Der Name des ersten Kapitels
$['kapitel'][0]['name']	Der Name des ersten Kapitels Ausführliche Schreibweise mit []
$.kapitel[*].id	Alle Ids der Kapitel
$..name	Alle Name Teile(Buch, Kapitel, Autor)
$.kapitel[?(@.name in 'Anfang', 'Ende')]	Alle Kapitel mit Namen Anfang oder Ende
$..*	Alle Elemente im Baum einzeln.

MT AG

data . cloud . mobile

Wolfgang Nast
Telefon: +49 2102 30961 – 0
wolfgang.nast@mt-ag.com