

JSON-Format

JSON ist ein Übertragungsformat für Daten, die sich leicht mit dem PC verarbeiten lassen sollen. Dabei gibt es Objekte mit Key-Value Paaren und Arrays von Objekten.

Objekte:

```
1 {
2   "key": "value",
3   "key2": 42,
4   "key3": true,
5   ...
6 }
```

Arrays:

```
1 [
2   {
3     "key": "objekt1",
4   },
5   {
6     "key": "objekt2"
7   },
8   ...
9 ]
```

Als Values in den Objekten sind zugelassen: Zahlen, Booleanwerte, Strings, oder ein weiteres Objekt oder sogar ein Array. Somit lassen sich die Objekte auch schachteln.

JSON lässt sich in Java mit folgenden Methoden verarbeiten:

- Relevante Methoden von JSONArray: `getJSONObject(int index)`
- Relevante Methoden von JSONObject: `getInt(String key)`, `getBoolean(String key)`, `getString(String key)`, `getJSONObject(String key)`, `getJSONArray(String key)`

Beispiel:

```
1 // Erstellt aus dem String ein JSONArray
2 JSONArray array = new JSONArray("[{ \"id\": 1 }]");
3
4 // Mit getJSONObject(int index) kann man auf die Objekte im Array zugreifen
5 JSONObject obj = array.getJSONObject(0);
6
7 // Mit getInt(String key) kann man fuer einen Key den Value auslesen
8 int zahl = obj.getInt("id");
```

Beispiel im SocialBotNet

```
1 public void eigenePinnwandLiken() {
2     // Alle Nachrichten auf der Pinnwand von MeinBot abrufen
3     String antwort = socialbotnet.GETAnfrageSenden("/api/pinnwand/MeinBot");
4     // Die Antwort ist ein JSONArray der Nachrichten
5     JSONArray nachrichten = new JSONArray(antwort);
6     // Iteriere ueber die einzelnen Nachrichten
7     for(int i=0; i<nachrichten.length(); i++) {
8         JSONObject nachricht = nachrichten.getJSONObject(i);
9         // Hole die Nachrichten ID aus dem JSONObject
10        int postid = nachricht.getInt("id");
11        // Diese ID wird als Eingabewert zum Liken benoetigt
12        liken(postid);
13    }
14 }
15
16 public void liken(int postid) {
17     // ... POST-Anfrage auf /api/like mit gesetzter postid
18 }
```

Beispielrückgaben des SocialBotNet

Ein *GET*-Request auf `/api/users` liefert folgendes Ergebnis:

```
1  [
2  {
3  {
4  "id":1,
5  "username":"root",
6  "hobbies":"Filme schauen, Fussball spielen",
7  "about":"I am root."
8  },
9  {
10 "id":2,
11 "username":"Welcome",
12 "hobbies": "",
13 "about": ""
14 },
15 ...
16 ]
```

Ein *GET*-Request auf `/api/users` oder `/api/pinnwand/...` liefert folgendes Ergebnis:

```
1  [
2  {
3  "id": 2,
4  "message": "Herzlich Willkommen, root!",
5  "publishingDate": "Oct 25, 2014 10:09:55 AM",
6  "user": {"id": 2, "username": "Welcome", "hobbies": "...", "about": "..."},
7  "wall": {"id": 1, "username": "root", "hobbies": "...", "about": "..."},
8  "trendingScore": 1.788332604357491,
9  "likes": 2,
10 "likedBy": [
11   {"id": 1, "username": "root", "hobbies": "...", "about": "..."},
12   {"id": 10, "username": "Hello", "hobbies": "...", "about": "..."}
13 ]
14 },
15 ...
16 ]
```
