

مجمع فنی طهران

JSON

Tehran Institute of Technology

Zahra Mansoori

z.mansoori@gmail.com

<http://tarsimm.com>

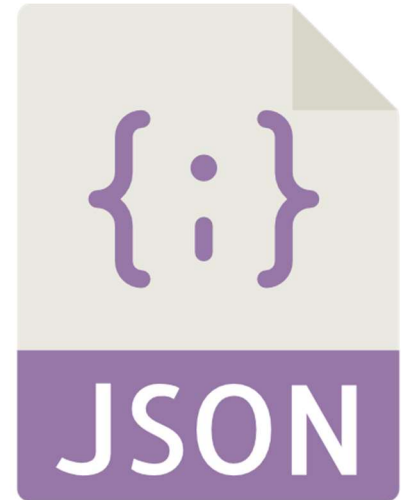
Contents

1. Introduction	2
2. Syntax.....	2
2.1. Simple data types.....	2
2.2. Composite data type.....	3
2.2.1. JSON Object.....	3
2.2.2. JSON Array	4

1. Introduction

JSON (JavaScript Object Notation) is one of the most popular and widely accepted data exchange format originally specified by Douglas Crockford.

- JSON is widely accepted in the software that includes client-server architecture for exchanging data between client and server
- JSON is easy to use and purely text-based, lightweight, and human-readable format and people often misunderstand as replacement of XML
- Although the abbreviation starts with JavaScript, JSON is not a language or have any language literals it just a specification for notation of data
- It is platform and language independent and inbuilt supported by almost all of the front line languages/frameworks like and support for the JSON data format is available in all the popular languages, some of which are C#, PHP, Java, C++, Python, Ruby and many more.
- The official Internet media type for JSON is application/json
- The JSON file name extension is .json



see also <http://json.org>

2. Syntax

A valid JSON expression can be one of the following data types:

Simple data types: String, Number, Boolean, Null

Composite data types: Value, Object, Array

2.1. Simple data types

<i>Data type</i>	<i>Examples of valid JSON</i>
<i>String</i>	"apple" "\u00c4pfel" ""
<i>Number</i>	3 1.4 -1.5e3
<i>Boolean</i>	true False
<i>Null</i>	null

2.2. Composite data type

<i>Data type</i>	<i>Examples of valid JSON</i>
<i>Value</i>	A JSON Value can be one of: String, Number, Boolean, Null, Object, Array
<i>Object</i>	Comma-separated unordered collection of names: Ex: <pre>{ "id": 1, "name": "A wooden door", "price": 12.50, "tags": ["home", "wooden"] }</pre> Or <pre>[1, 2, [3, 4, 5, 6], { "id": 1, "name": "A wooden door", "price": 12.50, "tags": ["home", "wooden"] }]</pre>
<i>Array</i>	Ordered collection of JSON values Ex: ["home", "wooden"]

2.2.1. JSON Object

A JSON Object is surrounded by curly braces and contains key-value pairs.

```
{ "key1": "value1", "key2": "value2", ... }
```

Keys must be valid strings, thus surrounded by double quotation marks. Values can be JSON objects, numbers, strings, arrays, or one of the following 'literal names': false, null, or true. In a key-value pair, the key is separated from the value by a colon. Multiple key-value-pairs are separated by commas.

Order in objects is not important. The following JSON object is thus equivalent to the above:

```
{ "key2": "value2", "key1": "value1", ... }
```

To sum it all up, this is an example of a valid JSON Object:

```
{
  "image": {
    "width": 800,
    "height": 600,
    "title": "View from 15th Floor",
    "thumbnail": {
      "url": "http://www.example.com/image/481989943",
      "height": 125,
      "width": 100
    },
    "visible": true,
    "ids": [116, 943, 234, 38793]
  }
}
```

2.2.2. JSON Array

A JSON Array is an ordered collection of values. It is surrounded by square braces i.e [], and values are comma-delimited:

```
{ "colors" : [ "red", "green", "blue" ] }
```

JSON Arrays can also contain any valid JSON element, including objects, as in this example of an array with 2 objects (taken from the RFC document):

```
[
  {
    "precision": "zip",
    "Latitude": 37.7668,
    "Longitude": -122.3959,
    "Address": "",
    "City": "SAN FRANCISCO",
    "State": "CA",
    "Zip": "94107",
    "Country": "US"
  },
  {
    "precision": "zip",
    "Latitude": 37.371991,
    "Longitude": -122.026020,
    "Address": "",
    "City": "SUNNYVALE",
    "State": "CA",
    "Zip": "94085",
  }
]
```

```
        "Country": "US"
    }
]
```

They can also contain elements with mixed types, for example:

```
[
    "red",
    51,
    true,
    null,
    {
        "state": "complete"
    }
]
```

A common mistake when writing JSON arrays (and objects) is to leave a trailing comma after the last element. This is common pattern in many languages, but unfortunately isn't valid in JSON. For example, the following array is invalid:

```
[
    1,
    2,
]
```