

Einführung in JSON

العربية / Български / 中文 / Český / Dansk / Nederlands / English / Esperanto / Français / Deutsch / Ελληνικά / עברית / Magyar / Indonesia / Italiano / 日本 / 한국어 / فارسی / Polski / Português / Română / Русский / Српско-хрватски / Slovenščina / Español / Svenska / Türkçe / Українська / Tiếng Việt

ECMA-404 The JSON Data Interchange Standard.

JSON (JavaScript Object Notation) ist ein schlankes Datenaustauschformat, das für Menschen einfach zu lesen und zu schreiben und für Maschinen einfach zu parsen (Analysieren von Datenstrukturen) und zu generieren ist. Es basiert auf einer Untermenge der **JavaScript Programmiersprache**, Standard ECMA-262 dritte Edition - Dezember 1999.

Bei JSON handelt es sich um ein Textformat, das komplett unabhängig von Programmiersprachen ist, aber vielen Konventionen folgt, die Programmieren aus der Familie der C-basierten Sprachen (inklusive C, C++, C#, Java, JavaScript, Perl, Python und vielen anderen) bekannt sind. Diese Eigenschaften machen JSON zum idealen Format für Datenaustausch.

JSON baut auf zwei Strukturen auf:

Name/Wert Paare. In verschiedenen Sprachen wird dies realisiert als ein Objekt (object), Satz (record), Struktur (struct), Wörterbuch bzw. Verzeichnis (dictionary), Hash-Tabelle (hash table), Schlüssel-Liste (keyed list) oder als ein assoziatives Array (associative array).

Eine geordnete Liste von Werten. In den meisten Sprachen wird das als

```

json
  element

value
  object
  array
  string
  number
  "true"
  "false"
  "null"

object
  '{' ws '}'
  '{' members '}'

members
  member
  member ' , ' members

member
  ws string ws ':' element

array
  '[' ws ']'
  '[' elements ']'
  
```

```
elements
  element
    element ' ' elements
```

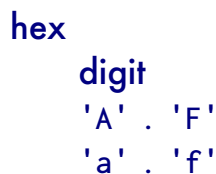
element
ws value ws

string
''' characters '''

```
characters
  ""
  character characters
```

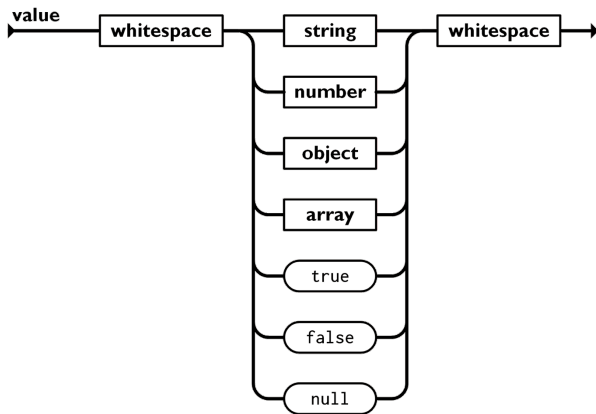


```
escape
    ' "'
    '\ '
    '/'
    'b'
    'f'
    'n'
    'r'
    't'
    'u' hex hex hex hex
```

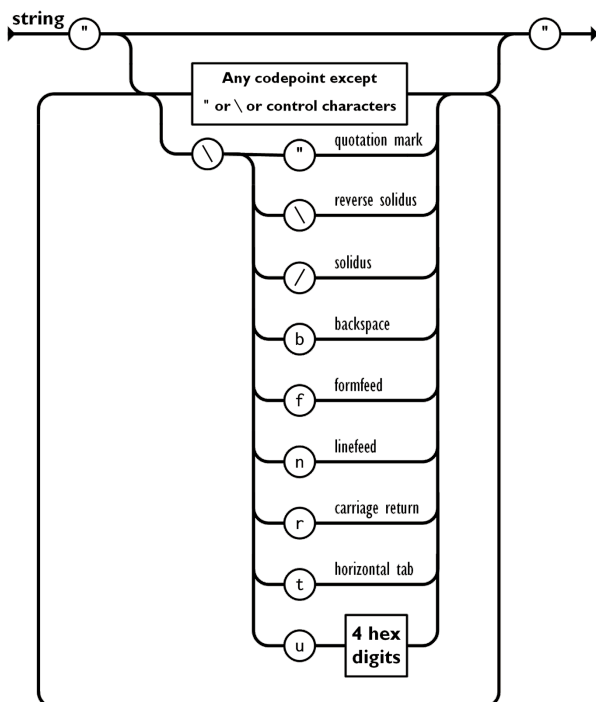


```
number
integer fraction exponent
```

```
integer
  digit
  onenine digits
  '-' digit
  '-' onenine digits
```



Eine **Zeichenkette** besteht aus keinem (leere Zeichenkette) oder mehr Unicode Zeichen und wird von doppelten Anführungszeichen umschlossen. Eine Zeichenkette kann Escape-Sequenzen mit einer besonderen Bedeutung enthalten. Ein einzelnes Zeichen wird durch eine Zeichenkette bestehend aus nur einem einzigen Zeichen dargestellt. Eine Zeichenkette (string) in JSON ist einer Zeichenkette in C oder Java sehr ähnlich.



Eine Zahl in JSON ist einer Zahl in C oder Java sehr ähnlich mit der Ausnahme, dass oktale und hexadezimale Zahlen nicht verwendet werden.

digits
 digit
 digit digits

digit
 '0'
 onenine

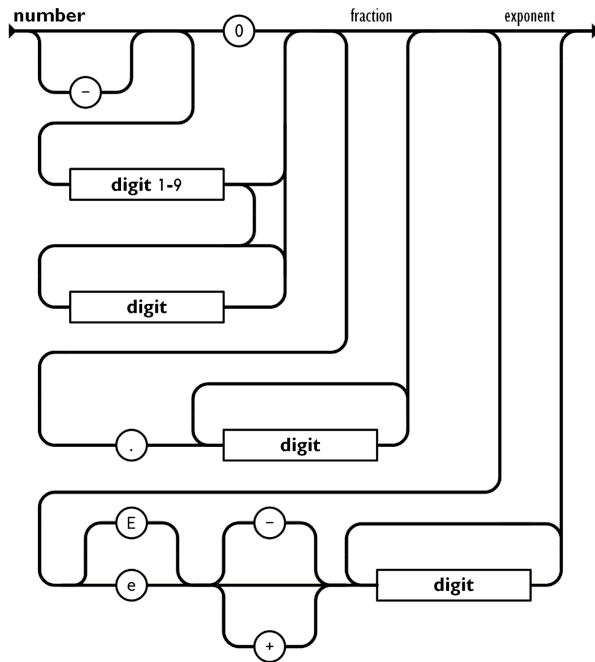
onenine
 '1' . '9'

fraction
 ""
 ' .' digits

exponent
 ""
 'E' sign digits
 'e' sign digits

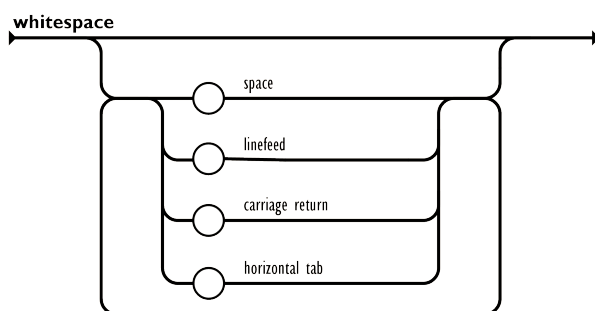
sign
 ""
 '+'
 '-'

ws
 ""
 '0020' ws
 '000A' ws
 '000D' ws
 '0009' ws



Leerzeichen können zwischen JSON-Elementen beliebig eingefügt werden.

It der Ausnahme einiger Details zur
Enkodierung beschreibt das die gesamte
Sprache.



8th

[json](#)

ActionScript

[ActionScript3](#)

Ada

[GNATCOLL.JSON](#)

AdvPL

[JSON-ADVPL](#)

APL

[JSON](#)

ASP

[JSON for ASP](#)

[JSON ASP utility class](#)

AWK

[JSON.awk](#)

ColdFusion

[SerializeJSON](#)

D

[std.json](#)

[asdf](#)

[vibe.data.json](#)

Dart

[json library](#)

Delphi

[Delphi Web Utils](#)

[JSON Delphi Library](#)

E

[JSON in TermL](#)

Erlang

[erl-json](#)

	rhawk	Fantom
BlitzMax		Json
	bmx-rjson	FileMaker
C		JSON
	JSON_checker	Fortran
	YAJL	json-fortran
	LibU	YAJL-Fort
	json-c	jsonff
	json-parser	Go
	jsonsl	package json
	WJElement	Groovy
	M's JSON parser	groovy-io
	cJSON	Haskell
	Jansson	RJson package
	jsmn	json package
	parson	Java
	ujson4c	JSON-java
	frozen	JSONUtil
	microjson	jsonp
	mjson	Json-lib
	progbase	Stringtree
	lwjson	SOJO
	cisson	json-taglib
C++		Flexjson
	JSONKit	Argo
	jsonme--	jsonij
	ThorsSerializer	fastjson
	JsonBox	mjson
	jvar	jjson
	rapidjson	json-simple
	JSON for Modern C++	json-io
	minijson	google-gson
	jsoncons	FOSS Nova JSON
	jsoncpp	Corn CONVERTER
	univalue	Apache johnzon
	ArduinoJson	Genson
	QJson	cookjson
	CAJUN	progbase
	libjson	jackson
	nosjob	MOXy
	JSON library for IoT	JavaScript
	qmjson	JSON
	JSON Support in Qt	json2.js
	JsonWax for Qt	clarinet
	progbase	Oboe.js
	Qentem-Engine	progbase
C#		LabVIEW
	fastJSON	flatten
	JSON_checker	Lisp
	Json.NET	Common Lisp JSON

JSON for .NET	LiveCode
Manatee Json	mergJSON
FastJsonParser	LotusScript
LightJson	JSON LS
Liersch.Json	Lua
Liersch.JsonSerialization	JSON Modules
progbase	M
JSON Essentials	DataBallet
Clojure	Matlab
data.json	JSONlab
Cobol	20565
Redvers COBOL JSON Interface	23393
Net.Data	
netdata-json	
Nim	
Module json	
Objective C	
NSJSONSerialization	
json-framework	
JSONKit	
yajl-objc	
TouchJSON	
OCaml	
jsonm	
PascalScript	
JsonParser	
Perl	
CPAN	
Photoshop	
JSON Photoshop Scripting	
PHP	
PHP 5.2	
PicoLisp	
picolisp-json	
Pike	
Public.Parser.JSON	
Public.Parser.JSON2	
PL/SQL	
pljson	
PureBasic	
JSON	
Puredata	
PuRestJson	
Python	
The Python Standard Library	
simplejson	
pyson	
Yajl-Py	
ultrajson	

metamagic.json
progbase

R

rjson
jsonlite

Racket

json-parsing

Rebol

json.r

RPG

JSON Utilities

Rust

Serde JSON
json-rust

Ruby

yajl-ruby
json-stream
progbase

Scala

circe

Scheme

MZScheme
JSON-struct

Shell

Jshon
JSON.sh
jwalk

Squeak

Squeak

Tcl

JSON

Visual Basic

VB-JSON
PW.JSON
.NET-JSON-Transformer
progbase

Visual FoxPro

fwJSON
JSON
vfpjson

- [Videos about JSON](#)
- [Videos about the JSON Logo](#)
- [Heresy & Heretical Open Source: A Heretic's Perspective](#)
- [How JavaScript Works](#) by Douglas Crockford