



JSON Schema Cheat Sheet

Terms

Instance - the JSON document being validated or described

Schema - the document containing the description

Constraints - created by adding validation keywords to the schema

type validation keyword values

string

integer

number - allows integer or floating point numbers

object

array

boolean

null

The type keyword can be either a string or an array e.g.

```
{ "type": "number" }  
{ "type": ["number", "string"] }
```

string type keywords

minLength - number

maxLength - number

pattern - regular expression in ECMA 262 syntax

format - date, date-time, uri, email, hostname, ipv4, ipv6, regex

integer / number type keywords

multipleOf - number

minimum - $x \geq$ number

exclusiveMinimum - $x >$ number

maximum - $x \leq$ number

exclusiveMaximum - $x <$ number

object type keywords

minProperties - number

maxProperties - number

required - array of property names

properties - map of property names to schemas

patternProperties - match property names with regex, validate values against a schema

additionalProperties - boolean - are properties not in the schema allowed?

dependencies - map dependencies between properties

propertyNames - validate property names against a schema

array type keywords

minItems - number

maxItems - number

uniqueItems - boolean - should all array items be unique?

items - schema or array of schemas to validate array items against

additionalItems - boolean or schema - false to disallow additional items, or schema for additional items

contains - schema - array must contain at least one item that validates against the schema

Keywords for all types

enum - array of items - data must deeply equal one item in the array

const - mixed type - restrict a value to a single value

title - string - short descriptive title

description - string - longer explanation about purpose of data described by the schema

default - mixed type - a default value

examples - array of examples that validate against the schema

\$comment - string - for adding a comment to the schema, not used for validation

Keywords for combining schemas

allOf - array - Must be valid against all of the subschemas

anyOf - array - Must be valid against any of the subschemas

oneOf - array - Must be valid against exactly one of the subschemas

not - object - Must not be valid against the given schema

if/then/else - Allow for conditional validation

This cheat sheet is based on draft-07 of JSON Schema (the most widely supported specification as of 2020-12) and the features implemented by Ajv.

Spotted a mistake? Drop an email to simon@simonplend.co.uk and let me know!

Sources

<https://json-schema.org/understanding-json-schema/>
<https://json-schema.org/specification-links.html#draft-7>
<https://ajv.js.org/keywords.html>

SP

Created by @simonplend

simonplend.com