



# JSON

javascript object notation

**tutorialspoint**

SIMPLY EASY LEARNING

[www.tutorialspoint.com](http://www.tutorialspoint.com)



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

## About the Tutorial

---

JSON or JavaScript Object Notation is a lightweight text-based open standard designed for human-readable data interchange. The JSON format was originally specified by Douglas Crockford, and is described in RFC 4627. The official Internet media type for JSON is application/json. The JSON filename extension is .json.

This tutorial will help you understand JSON and its use within various programming languages such as PHP, PERL, Python, Ruby, Java, etc.

## Audience

---

This tutorial has been designed to help beginners understand the basic functionality of JavaScript Object Notation (JSON) to develop the data interchange format. After completing this tutorial, you will have a good understanding of JSON and how to use it with JavaScript, Ajax, Perl, etc.

## Prerequisites

---

Before proceeding with this tutorial, you should have a basic understanding of the web application's work over HTTP and we assume that you have a basic knowledge of JavaScript.

## Copyright & Disclaimer

---

© Copyright 2017 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at [contact@tutorialspoint.com](mailto:contact@tutorialspoint.com)

## Table of Contents

---

About the Tutorial.....	i
Audience .....	i
Prerequisites .....	i
Copyright & Disclaimer.....	i
Table of Contents .....	ii
<b>1. JSON – OVERVIEW .....</b>	<b>1</b>
<b>Uses of JSON .....</b>	<b>1</b>
<b>Characteristics of JSON.....</b>	<b>1</b>
<b>Simple Example in JSON .....</b>	<b>1</b>
<b>2. JSON – SYNTAX.....</b>	<b>4</b>
<b>3. JSON – DATATYPES.....</b>	<b>5</b>
<b>Number.....</b>	<b>5</b>
<b>String .....</b>	<b>6</b>
<b>Boolean.....</b>	<b>7</b>
<b>Array .....</b>	<b>7</b>
<b>Object .....</b>	<b>8</b>
<b>Whitespace .....</b>	<b>8</b>
<b>null.....</b>	<b>9</b>
<b>JSON Value.....</b>	<b>9</b>
<b>4. JSON – OBJECTS.....</b>	<b>10</b>
<b>Creating Simple Objects .....</b>	<b>10</b>
<b>Creating Array Objects .....</b>	<b>11</b>

5.	JSON – SCHEMA .....	13
	JSON Schema Validation Libraries .....	13
	JSON Schema Example .....	14
6.	JSON – COMPARISON WITH XML.....	17
7.	JSON – JSON WITH PHP .....	18
	Environment .....	18
	JSON Functions.....	18
	Encoding JSON in PHP (json_encode) .....	18
	Decoding JSON in PHP (json_decode).....	19
8.	JSON – JSON WITH PERL.....	21
	Environment .....	21
	JSON Functions.....	21
	Encoding JSON in Perl (encode_json) .....	22
	Decoding JSON in Perl (decode_json) .....	23
9.	JSON – JSON WITH PYTHON .....	25
	Environment .....	25
	JSON Functions.....	25
	Encoding JSON in Python (encode).....	25
	Decoding JSON in Python (decode) .....	26
10.	JSON – JSON WITH RUBY .....	27
	Environment .....	27
	Parsing JSON using Ruby .....	27

11. JSON – JSON WITH JAVA.....	29
<b>Environment</b> .....	<b>29</b>
<b>Mapping between JSON and Java entities</b> .....	<b>29</b>
<b>Encoding JSON in Java</b> .....	<b>30</b>
<b>Decoding JSON in Java</b> .....	<b>31</b>
12. JSON – JSON WITH AJAX.....	33

# 1. JSON – OVERVIEW

JSON or JavaScript Object Notation is a lightweight text-based open standard designed for human-readable data interchange. Conventions used by JSON are known to programmers, which include C, C++, Java, Python, Perl, etc.

- JSON stands for JavaScript Object Notation.
- The format was specified by Douglas Crockford.
- It was designed for human-readable data interchange.
- It has been extended from the JavaScript scripting language.
- The filename extension is **.json**.
- JSON Internet Media type is **application/json**.
- The Uniform Type Identifier is public.json.

## Uses of JSON

---

- It is used while writing JavaScript based applications that includes browser extensions and websites.
- JSON format is used for serializing and transmitting structured data over network connection.
- It is primarily used to transmit data between a server and web applications.
- Web services and APIs use JSON format to provide public data.
- It can be used with modern programming languages.

## Characteristics of JSON

---

- JSON is easy to read and write.
- It is a lightweight text-based interchange format.
- JSON is language independent.

## Simple Example in JSON

---

The following example shows how to use JSON to store information related to books based on their topic and edition.

```
{
  "book": [
    {
      "id": "01",
      "language": "Java",
      "edition": "third",
      "author": "Herbert Schildt"
    },
    {
      "id": "07",
      "language": "C++",
      "edition": "second",
      "author": "E.Balagurusamy"
    }
  ]
}
```

After understanding the above program, we will try another example. Let's save the below code as **json.htm**:

```
<html>
<head>
<title>JSON example</title>
<script language="javascript" >

  var object1 = { "language" : "Java", "author" : "herbert schildt" };
  document.write("<h1>JSON with JavaScript example</h1>");
  document.write("<br>");
  document.write("<h3>Language = " + object1.language+"</h3>");
  document.write("<h3>Author = " + object1.author+"</h3>");

  var object2 = { "language" : "C++", "author" : "E-Balagurusamy" };
  document.write("<br>");
  document.write("<h3>Language = " + object2.language+"</h3>");
```

```
document.write("<h3>Author = " + object2.author+"</h3>");

document.write("<hr />");
document.write(object2.language + " programming language can be studied " +
"from book written by " + object2.author);
document.write("<hr />");

</script>
</head>
<body>
</body>
</html>
```

Now let's try to open json.htm using IE or any other javascript enabled browser that produces the following result:

## JSON with JavaScript example

  

**Language = Java**

**Author = herbert schildt**

  

**Language = C++**

**Author = E-Balagurusamy**

---

C++ programming language can be studied from book written by E-Balagurusamy

---

You can refer to JSON Objects chapter for more information on JSON objects.



## 2. JSON – SYNTAX

Let's have a quick look at the basic syntax of JSON. JSON syntax is basically considered as a subset of JavaScript syntax; it includes the following:

- Data is represented in name/value pairs.
- Curly braces hold objects and each name is followed by ':'(colon), the name/value pairs are separated by , (comma).
- Square brackets hold arrays and values are separated by ,(comma).

Below is a simple example:

```
{
  "book": [
    {
      "id": "01",
      "language": "Java",
      "edition": "third",
      "author": "Herbert Schildt"
    },
    {
      "id": "07",
      "language": "C++",
      "edition": "second",
      "author": "E.Balagurusamy"
    }
  ]
}
```

JSON supports the following two data structures:

- **Collection of name/value pairs:** This Data Structure is supported by different programming languages.
- **Ordered list of values:** It includes array, list, vector or sequence etc.

# 3. JSON – DATATYPES

JSON format supports the following data types:

Type	Description
Number	double- precision floating-point format in JavaScript
String	double-quoted Unicode with backslash escaping
Boolean	true or false
Array	an ordered sequence of values
Value	it can be a string, a number, true or false, null etc
Object	an unordered collection of key:value pairs
Whitespace	can be used between any pair of tokens
null	empty

## Number

---

- It is a double precision floating-point format in JavaScript and it depends on implementation.
- Octal and hexadecimal formats are not used.
- No NaN or Infinity is used in Number.

The following table shows the number types:

Type	Description
------	-------------

Integer	Digits 1-9, 0 and positive or negative
Fraction	Fractions like .3, .9
Exponent	Exponent like e, e+, e-,E, E+, E-

## Syntax

```
var json-object-name = {"string" : number_value, .....}
```

## Example

Example showing Number Datatype, value should not be quoted:

```
var obj = {"marks": 97}
```

## String

- It is a sequence of zero or more double quoted Unicode characters with backslash escaping.
- Character is a single character string i.e. a string with length 1.

The table shows various special characters that you can use in strings of a JSON document:

Type	Description
"	double quotation
\	backslash
/	forward slash
b	backspace
f	form feed
n	new line

r	carriage return
t	horizontal tab
u	four hexadecimal digits

## Syntax

```
var json-object-name = { string : "string value", ..... }
```

## Example

Example showing String Datatype:

```
var obj = {"name": "Amit"}
```

## Boolean

It includes true or false values.

## Syntax

```
var json-object-name = { string : true/false, ..... }
```

## Example

```
var obj = {"name": "Amit", "marks": 97, "distinction": true}
```

## Array

- It is an ordered collection of values.
- These are enclosed in square brackets which means that array begins with `[.` and ends with `].`
- The values are separated by `,` (comma).
- Array indexing can be started at 0 or 1.
- Arrays should be used when the key names are sequential integers.

## Syntax

```
[ value, .....]
```

## Example

Example showing array containing multiple objects:

```
{  
  "books": [  
    { "language":"Java" , "edition":"second" },  
    { "language":"C++" , "lastName":"fifth" },  
    { "language":"C" , "lastName":"third" }  
  ]  
}
```

End of ebook preview  
If you liked what you saw...  
Buy it from our store @ <https://store.tutorialspoint.com>