# **Summary JSON at Work Chapter One**

JavaScript Object Notation (JSON), is a data format. Currently many programming languages ​​and platforms have support for serializing and deserializing data of this nature. It was created in 2001 by Douglas Crockford and standardized in 2006 under the name RFC 4627.

Due to the resurgence of JavaScript in recent years as an environment language it has helped to make JSON popular. JSON is gradually gaining entry to XML as a format for data exchanges on the internet. The JSON format tends to have less overhead and is more compact, a developer no longer has to worry about arguing that it should be an Element or an Attribute. In its structure it can have constructions like objects, matrices and name / value pairs.

## **JSON Data Types (This title is copied from the JSON at Works book)**

* Name (or Key) / value pair
* Each name
* Is on the left side of the colon (:)
* Is a String, and must be surrounded by double quotes
* The value is to the right of the colon. In the preceding example, the value type is a String, but there are several other Value Types.
* Object
* Are enclosed within a beginning left curly brace ({) and an ending right curly brace (})
* Consist of comma-separated, unordered, name/value pairs.
* Can be empty, {}.
* Can be nested within other Objects or Arrays.
* Array
* Are enclosed within a beginning left brace ([) and an ending right brace (]).
* Consist of comma-separated, ordered values (see the next section).
* Can be empty, [].
* Can be nested within other Arrays or Objects.
* Have indexing that begins at 0 or 1.
* String
* • Strings consist of zero or more Unicode characters enclosed in quotation marks (""). Please see the following list for additional valid characters.
* Strings wrapped in single quotes (') are not valid.
* Number
* Numbers are always in base 10 (only digits 0–9 are allowed) with no leading zeros.
* Numbers can have a fractional part that starts with a decimal point (.).
* Numbers can have an exponent of 10, which is represented with the e or E notation with a plus or minus sign to indicate positive or negative exponentiation.
* Octal and hexadecimal formats are not supported.
* Unlike JavaScript, numbers can’t have a value of NaN (not a number for invalid numbers) or Infinity.
* Boolean
* Booleans can have a value of only true or false.
* The true or false value on the righthand side of the colon (:) is not surrounded by quotes.
* Null
* Are not surrounded by quotes
* Indicate that a key/property has no value
* Act as a placeholder

## **JSON Comments**

There are no comments in a JSON document.

## **JSON Property Names**

Two styles are allowed to format property names (lowerCamelCase and snake\_case)