**JSON** stands for **J**ava**S**cript **O**bject **N**otation. It is a format for structuring data. This format is used by different web applications to communicate with each other. JSON is the replacement of the XML data exchange format in JSON. It is easy to struct the data compare to XML. It supports data structures like arrays and objects and the JSON documents that are rapidly executed on the server. It is also a Language-Independent format that is derived from JavaScript. The official media type for the JSON is application/json and to save those file **.json** extension.

**Features of JSON:**

* **Easy to understand:** JSON is easy to read and write.
* **Format:** It is a text-based interchange format. It can store any kind of data in an array of video, audio, and image anything that you required.
* **Support:** It is light-weighted and supported by almost every language and OS. It has a wide range of support for the browsers approx. each browser supported by JSON.
* **Dependency:** It is an Independent language that is text-based. It is much faster compared to other text-based structured data.

**JSON Syntax Rules:** Data is in name/value pairs and they are separated by commas. It uses curly brackets to hold the objects and square brackets to hold the arrays.

**Example:**

{

"name": "cppsecrets",

"site": "https://cppsecrets.com/",

"sections" : ["C++", "C++ Algorithms","Python",

"Python Django","GDB","Linux"]  
 }

**Advantages of JSON:**

* JSON stores all the data in an array so data transfer makes easier. That’s why JSON is the best for sharing data of any size even audio, video, etc.
* Its syntax is very easy to use. Its syntax is very small and light-weighted that’s the reason that it executes and response in a faster way.
* JSON has a wide range for the browser support compatibility with the operating systems, it doesn’t require much effort to make it all browser compatible.
* On the server-side parsing the most important part that developers want, if the parsing will be fast on the server side then the user can get the fast response, so in this case JSON server-side parsing is the strong point compare tot others.

**Disadvantages of JSON:**

* The main disadvantage for JSON is that there is no error handling in JSON, if there was a slight mistake in the JSON script then you will not get the structured data.
* JSON becomes quite dangerous when you used it with some unauthorized browsers. Like JSON service return a JSON file wrapped in a function call that has to be executed by the browsers if the browsers are unauthorized then your data can be hacked.
* JSON has limited supported tools that we can use during JSON development.

**Json library in C++ was developed by Niels Lohmann.**

Note: You need to set necessary switches to enable C++11(e.g.,-std=c++11 for GCC).

In order to work with the library, you need to first download source code and include it into your project. (Download the zip file). Link for source code: <https://github.com/nlohmann/json>

You can either include "include/nlohmann" folder or "single\_include/nlohmann" folder into your compiler options. Now, you can use json library by including it as a header file into your project.

The advantages of using this library are:

* It is very easy to include into your project as it consists of only single header.
* A clean and simple API without excessively verbose syntax.
* Good test coverage including Exceptional Handling.

**Usage in C++ Application:**

**#include <nlohmann/json.hpp>**

**using json = nlohmann::json;**

If your goal is to speed up your development by adding JSON support, then this is the best library you can choose.

**Installation Steps:**

**$ git clone https://github.com/nlohmann/json.git**

**$ cd json**

**$ mkdir build**

**$ cd build**

**$ cmake ..**

**$ make -j4**

**$ sudo make install**