JSON is an extremely simple data-interchange format. As json.org says, it is easy for humans to read and write and for machines to parse and generate.

A piece of JSON must represent either: * A collection of name/value pairs (`{ }`). In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array. * An ordered list of values (`[ ]`). In various languages, this is realized as an array, vector, list, or sequence. an array/list/sequence (`[ ]`) or a dictionary/object/associated array (`{ }`).

JSON in its purest form has no actual comments, but most parsers will accept C-style (`//`, `/* */`) comments. Some parsers also tolerate a trailing comma (i.e. a comma after the last element of an array or the after the last property of an object), but they should be avoided for better compatibility.

For the purposes of this tutorial, everything is going to be 100% valid JSON. Luckily, it kind of speaks for itself.

Supported data types:

- Strings: `"hello"`, `"\"A quote.\""`, `"\u0abe"`, `"Newline.\n"`
- Numbers: `23`, `0.11`, `12e10`, `3.141e-10`, `1.23e+4`
- Objects: `{ "key": "value" }`
- Arrays: `["Values"]`
- Miscellaneous: `true`, `false`, `null`

```json
{
  "key": "value",

  "keys": "must always be enclosed in double quotes",
  "numbers": 0,
  "strings": "Hellø, wørld. All unicode is allowed, along with \"escaping\".",
  "has bools?": true,
  "nothingness": null,

  "big number": 1.2e+100,

  "objects": {
    "comment": "Most of your structure will come from objects.",

    "array": [0, 1, 2, 3, "Arrays can have anything in them.", 5],

    "another object": {
      "comment": "These things can be nested, very useful."
    }
  },

  "silliness": [
    {
      "sources of potassium": ["bananas"]
    },
    [
      [1, 0, 0, 0],
      [0, 1, 0, 0],
      [0, 0, 1, "neo"],
      [0, 0, 0, 1]
    ]
  ],
```

```json
  "alternative style": {
    "comment": "check this out!"
  , "comma position": "doesn't matter, if it's before the next key, it's valid"
  , "another comment": "how nice"
  },



  "whitespace": "Does not matter.",



  "that was short": "And done. You now know everything JSON has to offer."
}
```

Further Reading

- JSON.org All of JSON beautifully explained using flowchart-like graphics.