

JavaScript Object Notation (JSON)

- JSON is a lightweight data-interchange format that is completely language independent.
- It was derived from JavaScript, but many modern programming languages include code to generate and parse JSON-format data
- The official Internet media type for JSON is application/json.
- It was designed for human-readable data interchange.
- The filename extension is .json.



Uses of JSON

- It is used while writing JavaScript based applications that includes browser extensions and websites.
- JSON format is used for serialising 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.



Characteristics of JSON

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



Understanding JSON Structure

```
{
  "Title": "The Cuckoo's Calling",
  "Author": "Robert Galbraith",
  "Genre": "classic crime novel",
  "Detail": {
    "Publisher": "Little Brown",
    "Publication_Year": 2013,
    "ISBN-13": 9781408704004,
    "Language": "English",
    "Pages": 494
  },
  "Price": [
    {
      "type": "Hardcover",
      "price": 16.65,
    },
    {
      "type": "Kindle Edition",
      "price": 7.03,
    }
  ]
}
```

Diagram illustrating the JSON structure with annotations:

- Object Starts**: Points to the opening curly brace `{` at the beginning of the root object.
- Object Starts**: Points to the opening curly brace `{` inside the `"Detail"` object.
- Value string**: Points to the string value `"Little Brown"` for the `"Publisher"` key.
- Value number**: Points to the numeric value `2013` for the `"Publication_Year"` key.
- Object ends**: Points to the closing curly brace `}` of the `"Detail"` object.
- Array starts**: Points to the opening square bracket `[` of the `"Price"` array.
- Object Starts**: Points to the opening curly brace `{` of the first object in the `"Price"` array.
- Object ends**: Points to the closing curly brace `}` of the first object in the `"Price"` array.
- Object Starts**: Points to the opening curly brace `{` of the second object in the `"Price"` array.
- Object ends**: Points to the closing curly brace `}` of the second object in the `"Price"` array.
- Array ends**: Points to the closing square bracket `]` of the `"Price"` array.
- Object ends**: Points to the closing curly brace `}` of the root object.



JSON – Data Types

- **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



Bad Special Characters Inside JSON & Solution

- **Backspace** : Replace with `\b`
- **Form feed** : Replace with `\f`
- **Newline** : Replace with `\n`
- **Carriage return** : Replace with `\r`
- **Tab** : Replace with `\t`
- **Double quote** : Replace with `\"`
- **Backslash** : Replace with `\\`



Best practices of JSON

- Always enclose the Key : Value pair within double quotes

```
{'id': '1','name':File} is not right X
```

```
{"id": 1,"name":"File"} is okay ✓
```

```
{"id": "1","name":"File"} is the best ✓
```



Best practices of JSON

- Never use Hyphens in your Key fields

```
{"first-name":"Rachel","last-name":"Green"} is not right. X
```

```
{"first_name":"Rachel","last_name":"Green"} is okay ✓
```

```
{"firstname":"Rachel","lastname":"Green"} is okay ✓
```

```
{"firstName":"Rachel","lastName":"Green"} is the best. ✓
```



Best practices of JSON

- Bad Special Characters And Solution

```
"<h2>About US <br>\r\n</h2>\r\n<p>It has been exactly 3 years since I wrote my first blog series
\r\nentitled "Flavorful Tuscany", but starting it was definitely not easy. \r\nBack then, I didn't know
much about blogging, let alone think that one \r\n day it could become <strong>my full-time job</strong>.
Even though I had\r\n many recipes and food-related stories to tell, it never crossed my mind\r\n that I
could be sharing them with the whole world.</p>\r\n<p>I am now a <strong>full-time blogger</strong> and
the curator of the <a data-cke-saved-href=\r\n"\"https://ckeditor.com/ckeditor-4/#\"
href=\r\n"\"https://ckeditor.com/ckeditor-4/#\">Simply delicious newsletter</a>, sharing stories about
traveling and cooking, as well as tips on how to run a successful blog.</p>\r\n<p>If you are tempted by
the idea of creating your own blog, please think about the following:</p>\r\n<ul><li>Your story (what do
you want to tell your audience)</li><li>Your audience (who do you write for)</li><li>Your blog name and
design<br>\r\n\t</li></ul>\r\n<p>After you get your head around these 3 essentials, all you have to do
is grab your keyboard and the rest will follow.</p>"
```



Best practices of JSON

- Always create a Root element.

JSON with root element

```
{
  "menu": [
    {
      "id": "1",
      "name": "File",
      "value": "F",
      "popup": {
        "menuitem": [
          {"name": "New", "value": "1N", "onclick": "newDoc()"},
          {"name": "Open", "value": "1O", "onclick": "openDoc()"},
          {"name": "Close", "value": "1C", "onclick": "closeDoc()"}
        ]
      }
    },
    {
      "id": "2",
      "name": "Edit",
      "value": "E",
      "popup": {
        "menuitem": [
          {"name": "Undo", "value": "2U", "onclick": "undo()"},
          {"name": "Copy", "value": "2C", "onclick": "copy()"},
          {"name": "Cut", "value": "2T", "onclick": "cut()"}
        ]
      }
    }
  ]
}
```

JSON without root element

```
[
  {
    "id": "1",
    "name": "File",
    "value": "F",
    "popup": {
      "menuitem": [
        {"name": "New", "value": "1N", "onclick": "newDoc()"},
        {"name": "Open", "value": "1O", "onclick": "openDoc()"},
        {"name": "Close", "value": "1C", "onclick": "closeDoc()"}
      ]
    }
  },
  {
    "id": "2",
    "name": "Edit",
    "value": "E",
    "popup": {
      "menuitem": [
        {"name": "Undo", "value": "2U", "onclick": "undo()"},
        {"name": "Copy", "value": "2C", "onclick": "copy()"},
        {"name": "Cut", "value": "2T", "onclick": "cut()"}
      ]
    }
  }
]
```

