Theory: JSON

🕓 13 minutes 0 / 5 problems solved

Skip this topic Start practicing

4399 users solved this topic. Latest completion was **about 1 hour ago**.

JSON (or JavaScript Object Notation) is a text-based format for storing and transmitting structured data. It comes from the JavaScript language, but it is still considered to be language-independent: it works with almost any programming language. With JSON's lightweight syntax, you can easily store and send to other apps everything from numbers and strings to arrays and objects. You can also create more complex data structures by linking arrays to each other.

§1. Basic syntax and structure

JSON text can be built on one of two structures:

- a collection of key:value pairs (associative array);
- an orderly set of values (array or list).

JSON objects are written in curly braces `{}`, and their key:value pairs are separated by a comma `,`. The key and the value in the pair are separated by a colon `:`. Here is an example for you:

```
{
    "first_name": "Sophie",
    "last_name": "Goodwin",
    "age": 34
}
```

Here you can see some user's data in JSON format.

Keys in an object are always strings, but values can be any of seven types of values, including another object or array.

Arrays are written in square brackets `[]` and their values are separated by a comma `,`. The value in the array, again, can be of any type, including another array or object. Here is an example of an array:

```
["night", "street", false, [ 345, 23, 8, "juice"], "fruit"]
```

Most often, an array will include similar elements.

JSON does not support comments.

§2. Nested objects

JSON is a highly flexible format. You can nest objects inside other objects as properties:

Current topic:

JSON ···

Topic is required for:

Postman ···

HTTP clients ···

Rest controller ···

Json module ···

Table of contents:

```
{
  "persons": [
    {
      "firstName": "Whitney",
      "lastName": "Byrd",
      "age": 20
    },
    {
      "firstName": "Eugene",
      "lastName": "Lang",
      "age": 26
    },
    {
      "firstName": "Sophie",
      "lastName": "Goodwin",
      "age": 34
    }
  ]
}
```

If objects and arrays contain other objects or arrays, the data has a tree-like structure.

The nested objects are fully independent and may have different properties:

```
{
  "persons": [
    {
      "firstName": "Whitney",
      "age": 20
    },
    {
      "firstName": "Eugene",
      "lastName": "Lang"
    }
  ]
}
```

But in practice, such objects often look similar.

§3. camelCase VS under_score

If you have read the JSON objects examples really carefully, you might have a lingering question: what style of compound word writing should be used for JSON?

CamelCase is a style where compound words are written together and without spaces, but each word inside the phrase starts with a capital letter. The style is called camelCase because the capital letters inside the word resemble camel's humps.

In underscore style, compound words are written through the bottom underline.

> In fact, the choice of the right JSON naming convention depends directly on your programming language and libraries. You can use both camelCase and Underscore, any choice will be valid, but do not mix them together in one JSON.

§4. The advantages of JSON

JSON is widely spread for data exchange on the Internet because of its strong advantages:

- compactness;
- flexibility;
- high readability, even for people far from programming;
- most programming languages have functions and libraries for reading and creating JSON structures.

The JSON is a general format to pass structured data through the network because after you serialize data to JSON, you can deserialize it back without losing any information. The main advantage of JSON comparing to plain text is the ability to describe relations between objects via nesting and key-value pairs. So, it's high chances that the sites you're often visiting use JSON too.

Other popular applications of JSON are data storage and configuration files for other programs.

§5. Afterword

Now you have seen that JSON is easy to understand and use, and it's quite awesome since it's a very useful tool for transferring data between applications. In working practice, you probably won't have to create JSON files yourself, you will get them from other sources, but if you want to save the code on your computer, you should save the files in the `.json` extension.

Report a typo

477 users liked this theory. 6 didn't like it. What about you?

😍 🙂 😐 🙁 😡

Start practicing

Comments (16) Hints (0) Useful links (0) Show discussion