

JS解析JSON

JSON 全称为“JavaScript Object Notation”，是当前最流行的一种轻量级的数据交换格式，用来存储和传输数据，通常服务器端与客户端在进行交互时就是使用 JSON 格式的数据。

什么是 JSON

JSON 是从 JavaScript 脚本中演变而来的，并且使用 JavaScript 中的语法来描述数据。与 XML 一样，JSON 也是一种基于文本的格式，文件扩展名为 `.json`。

目前大多数编程语言（例如 PHP、Java、.Net）都支持 JSON，JSON 中有两种基本的结构：

- 对象：由若干键/值对（即 `key:value`）组成的无序集合，每个对象以左花括号 `{` 开始，以右花括号 `}` 结尾，多个键/值对之间使用逗号 `,` 分隔；
- 数组：一个有序的值列表，每个数组以左方括号 `[` 开始，以右方括号 `]` 结尾，多个值之间使用逗号 `,` 分隔。

在 JSON 中，属性名称或键都是字符串格式的（需要使用英文的双引号括起来），而值则可以是任意类型，如下所示：

```
{
  "course": {
    "name": "JavaScript",
    "author": "http://c.biancheng.net/",
    "year": 2021,
    "genre": "Getting Started tutorial",
    "bestseller": true
  },
  "fruits": [
    "Apple",
    "Banana",
    "Strawberry",
    "Mango"
  ]
}
```

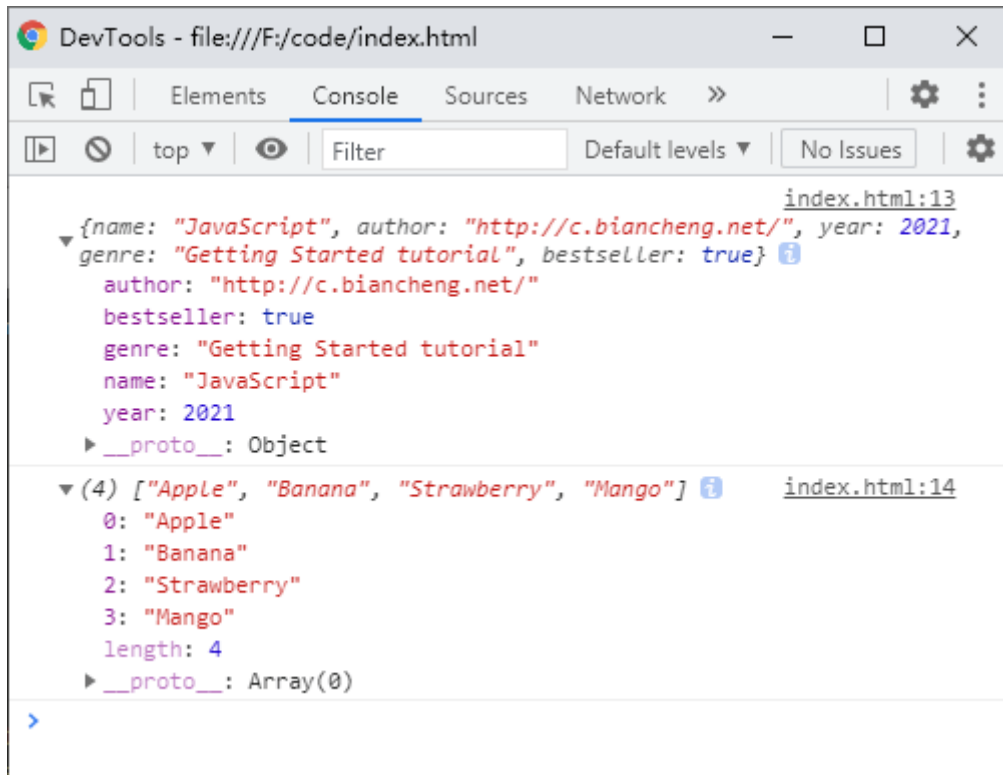
在 JavaScript 中解析 JSON 数据

在 JavaScript 中，您可以使用 `JSON.parse()` 方法来解析 JSON 数据，示例代码如下：

```
01. var json = '{"course": {"name": "JavaScript", "author":
    "http://c.biancheng.net/", "year": 2021, "genre": "Getting Started
    tutorial", "bestseller": true}, "fruits":
    ["Apple", "Banana", "Strawberry", "Mango"]}';
02. var obj = JSON.parse(json);
```

```
03. console.log(obj.course);
04. console.log(obj.fruits);
```

允许结果如下图所示：



图：JSON.parse() 方法演示

解析嵌套的 JSON 数据

JSON 数据中的对象和数组可以相互嵌套，一个 JSON 对象中可以包含任意类型的数据（例如数组、嵌套数组、其它 JSON 对象等）。对于相互嵌套的 JSON 数据我们要如何获取呢？示例代码如下：

```
01. var json = `{
02.     "book": {
03.         "name": "Harry Potter and the Goblet of Fire",
04.         "author": "J. K. Rowling",
05.         "year": 2000,
06.         "characters": ["Harry Potter", "Hermione Granger", "Ron Weasley"],
07.         "genre": "Fantasy Fiction",
08.         "price": {
09.             "paperback": "$10.40", "hardcover": "$20.32", "kindle": "$4.11"
10.         }
11.     }
12. }`;
13.
14. // 将 JSON 数据转换为 JSON 对象
15. var obj = JSON.parse(json);
16.
```

```
17. // 打印嵌套的 JSON 数据
18. function printValues(obj) {
19.     for (var k in obj) {
20.         if (obj[k] instanceof Object) {
21.             printValues(obj[k]);
22.         } else {
23.             document.write(obj[k] + "<br>");
24.         };
25.     }
26. };
27.
28. // 调用 printValues() 函数
29. printValues(obj);
30.
31. document.write("<hr>");
32.
33. // 打印 JSON 数据中的单个值
34. document.write(obj["book"]["author"] + "<br>");           // 输出: J. K.
    Rowling
35. document.write(obj["book"]["characters"][0] + "<br>");     // 输出: Harry
    Potter
36. document.write(obj["book"]["price"]["hardcover"]);       // 输出: $20.32
```

运行结果如下图所示:

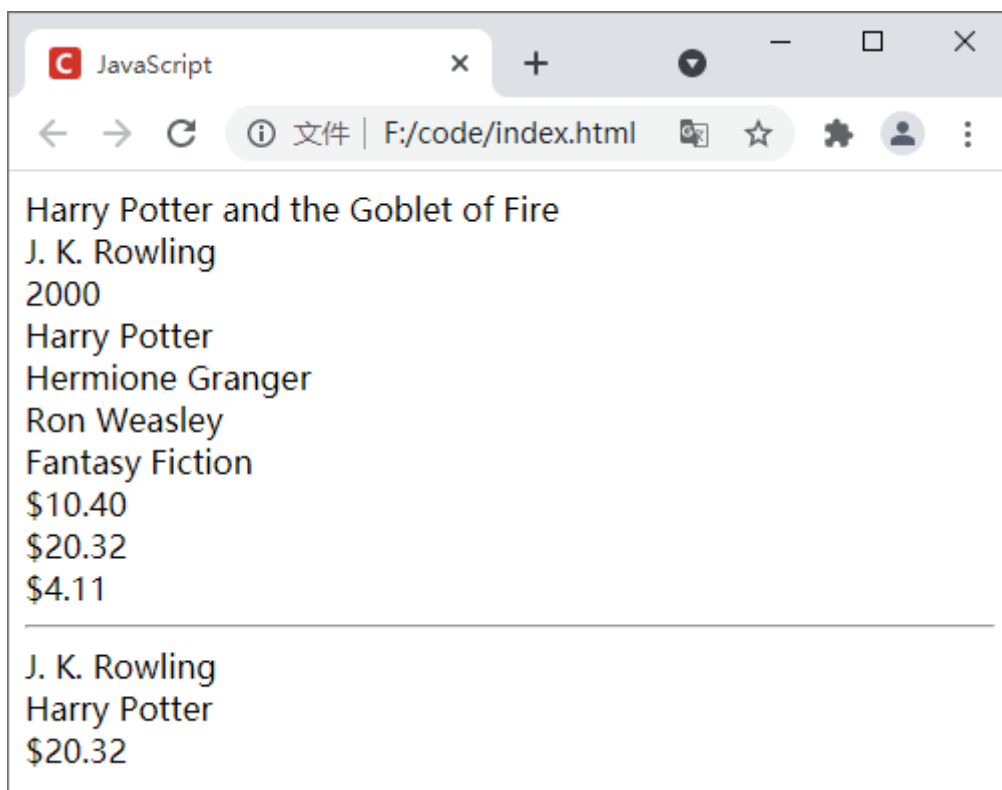


图: 解析嵌套的 JSON 数据

将数据转换为 JSON

在开发过程中，有时我们需要将数据转换为 JSON 格式，方便客户端与服务器端进行数据交互。JavaScript 中提供了 `JSON.stringify()` 方法来将 JavaScript 值转换为 JSON 格式，如下例所示：

```
01.  var obj = {  
02.      "name": "JavaScript",  
03.      "author": "http://c.biancheng.net/",  
04.      "year": 2021,  
05.      "genre": "Getting Started tutorial",  
06.      "bestseller": true  
07.  };  
08.  var json = JSON.stringify(obj);  
09.  document.write(json);
```

运行结果如下：

```
{"name":"JavaScript","author":"http://c.biancheng.net/","year":2021,"genre":"Getting Started  
tutorial","bestseller":true}
```

注意：虽然 JavaScript 对象与 JSON 对象看起来非常相似，但它们并不相同，例如在 JavaScript 中，对象的属性名称可以用单引号 `'` 或双引号 `"` 括起来，也可以完全省略引号。但是，在 JSON 中，所有属性名称都必须用双引号括起来。