

Работа с JSON

Урок 10





Кадочников Алексей

Frontend-разработчик

- ☀ Веб-разработчик со стажем более 9 лет
- ☀ Преподаватель GeekBrains с 2015 года
- ☀ Автор курсов по Frontend на портале Geekbrains
- ☀ Работал в таких компаниях, как VK и Wizard-C



План курса

1

Знакомство с JavaScript

2

Основы JavaScript

3

Функции в JavaScript

4

Циклы и массивы

5

Работа с объектами в JavaScript

6

Введение в DOM

7

Работа с DOM

8

Основы событий в JavaScript

9

Работа с событиями в JavaScript

10

Основы шаблонизации, работа с JSON

11






Работа с медиа файлами

12

Основы API, итоги курса



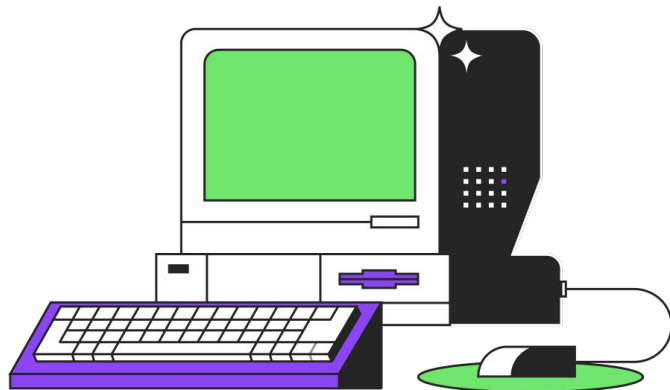
Что будет на уроке сегодня

-  Введение в JSON
-  Синтаксис и структура
-  Работаем с комплексными типами в JSON
-  Функции в JSON
-  Заключение



Введение в JSON

JSON (сокращение от JavaScript Object Notation) — это формат передачи данных. Он доступен для использования во многих языках, включая Python, Ruby, PHP и Java.





Введение в JSON

В программах на JavaScript формат JSON обычно используется в следующих случаях:

1. Хранение данных.
2. Генерирование структур данных из пользовательского ввода.
3. Обмен данными между сервером и клиентом.
4. Настройка и проверка данных.





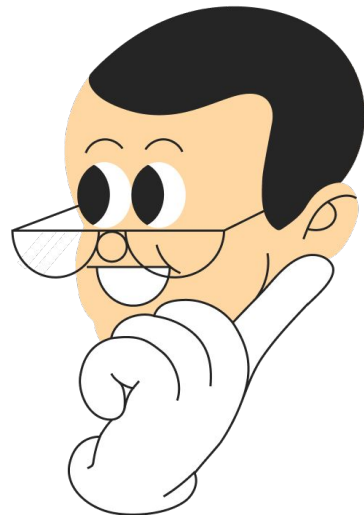
Синтаксис и структура





Синтаксис и структура

Объект JSON содержит данные в формате «ключ-значение» и обычно заключаются в фигурные скобки.





Синтаксис и структура

Вот так выглядит JSON-объект:

```
1 {  
2   "first_name": "Sammy",  
3   "last_name": "Shark",  
4   "location": "Ocean",  
5   "online": true,  
6   "followers": 987  
7 }  
8
```



Синтаксис и структура

Пары ключ-значение разделены двоеточием.

Пример: "key" : "value"

Друг от друга такие пары отделены запятыми. Таким образом, середина JSON-объекта выглядит так:

"key" : "value",

"key" : "value",

"key" : "value"





Синтаксис и структура

Хотя в файлах .json мы обычно видим данные, разнесёнными на несколько строк, JSON также может быть записан в одну сплошную строку:

```
1 { "first_name": "Sammy", "last_name": "Shark", "online": true, }
```

Такой подход наиболее распространён в файлах других форматов или при работе с JSON-строкой.



Синтаксис и структура

Работа с JSON в многострочном формате зачастую делает код более удобочитаемым, особенно когда вы пытаетесь справиться с большим набором данных.

```
1 {  
2   "first_name": "Sammy",  
3   "last_name": "Shark",  
4   "online": true  
5 }  
6
```



Синтаксис и структура

Очень важно помнить, что похожие визуально объекты JSON и объекты JavaScript на самом деле отличаются по формату. И хотя вы можете использовать функции внутри JavaScript-объектов, вы не можете использовать их как значения в JSON.





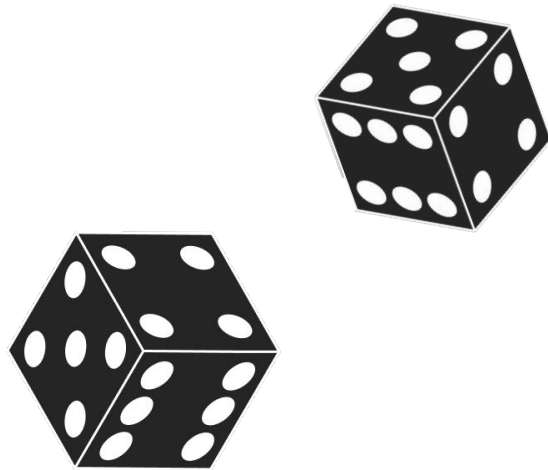
Работаем
с комплексными типами
в JSON





Работаем с комплексными типами в JSON

В дополнение ко вложенным массивам, JSON может содержать и другие вложенные объекты. Такие объекты и массивы будут передаваться как значения, присвоенные ключам, и будут представлять собой связку ключ-значение.





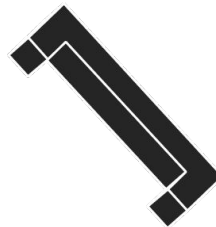
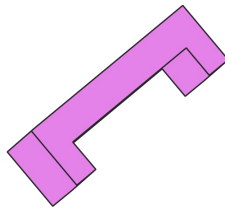
Вложенные объекты

```
1 {  
2   "sammy": {  
3     "username": "SammyShark",  
4     "location": "Indian Ocean",  
5     "online": true,  
6     "followers": 987  
7   },  
8   "jesse": {  
9     "username": "JesseOctopus",  
10    "location": "Pacific Ocean",  
11    "online": false,  
12    "followers": 432  
13  }  
14 }
```




Вложенные массивы

Также в формате JSON данные могут быть вложены с использованием массивов JavaScript в качестве значений. JavaScript использует квадратные скобки [] для формирования массива.





Вложенные массивы

Пример:

```
1
2 {
3   "first_name": "Sammy",
4   "last_name": "Shark",
5   "location": "Ocean",
6   "websites": [{
7     "description": "work",
8     "URL": "https://www.digitalocean.com/"
9   },
10  {
11    "description": "tutorials",
12    "URL": "https://www.digitalocean.com/community/tutorials"
13  }
14 ],
15
```



Функции в JSON





Функции в JSON

При работе с JSON очень полезно иметь возможность быстро преобразовать строку в объект и наоборот. В этом разделе мы рассмотрим два метода JSON.

- Функция `JSON.stringify()`
- Функция `JSON.parse()`





Функция JSON.stringify()

Эта функция преобразовывает объекты JSON в строки.

Рассмотрим объект, присвоенный переменной obj. Попробуйте преобразовать его в строку. Для этого нужно передать функции JSON.stringify() переменную obj. Присвойте эту строку переменной s.

```
1 const obj = {"first_name" : "John", "last_name" : "Smith",  
  "location" : "London"}  
2 const s = JSON.stringify(obj)  
3
```



Функция `JSON.parse()`

Строки удобны при обмене данными, но потом их нужно снова преобразовать в объекты. Для этого используют функцию `JSON.parse()`.

Примечание: чтобы преобразовать текст в объект, используйте функцию `eval()`.

Теперь попробуйте преобразовать значение функции `s` в объект и присвоить его новой переменной:

```
1 const data = JSON.parse(s);
```

Получаем объект `data`, идентичный объекту `obj`.



Пример JSON.parse()



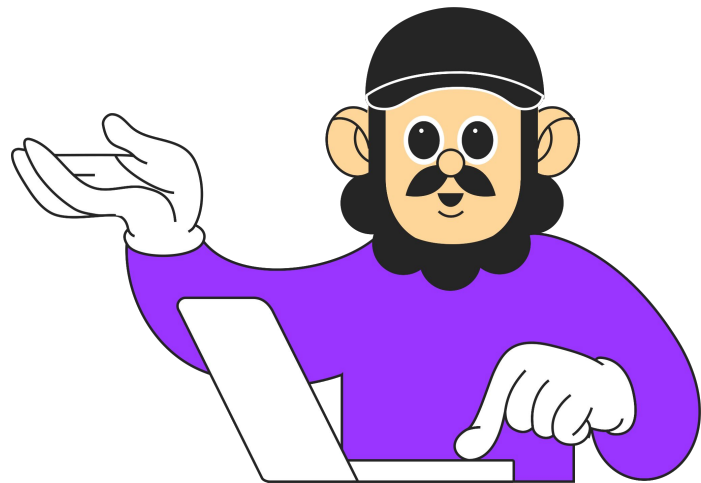
Заключение










Заключение


JSON — простой формат, который позволяет вам легко делиться данными, хранить их и работать с ними. Как формат, JSON переживает растущую поддержку API.





Итоги урока

-  Введение в JSON
-  Синтаксис и структура
-  Работаем с комплексными типами в JSON
-  Функции в JSON
-  Заключение

Спасибо 
за внимание

