**Формат JSON, Вебкадеми**

**JSON** - это формат предоставления данных в виде строки. Используется для передачи данных от backend к frontend и обратно. Может использоваться для глубокого клонирования несложных структур данных.

**JSON** поддерживает следующие типы данных:

* Объекты { ... }
* Массивы [ ... ]
* Примитивы:
* строки,
* числа,
* логические значения true/false,
* null.

**Строки** используют **двойные кавычки**. Никаких одинарных кавычек или обратных кавычек в JSON. **Строковые ключи** объекта также заключаются в **двойные кавычки**. В отличие от структур JavaScript, в JSON не разрешены запятые после последних элементов массивов и объектов

## Правила написания JSON файла

1. Описываем как объект или как массив.
2. Названия полей в объекте заключаем в кавычки.
3. Значения в виде строк заключаем в кавычки.
4. Используем только **двойные кавычки** "".
5. Значение в виде чисел или true / false в кавычки не заключаем
6. Значения разделяем ","
7. После последнего значения (свойства в объекте или элемента в массиве "," не ставится)

**!** Эти правила строгие. При ошибке вся строка не JSON. Даже одна неуместная запятая или двоеточие могут привести к сбою JSON-файла и не работать. Вы должны быть осторожны, чтобы проверить любые данные, которые вы пытаетесь использовать (хотя сгенерированный компьютером JSON с меньшей вероятностью включает ошибки, если программа генератора работает правильно).

Вы можете проверить JSON с помощью приложения вроде **JSONLint**.

**!** JSON - это чисто формат данных - он содержит только свойства, без методов. JSON не кодирует функции и методы в объектах. Они будут пропущены.

const personWithMethod = {

name: "John",

age: 26,

isMarried: true,

**sayHello**: function () {

console.log("Hello!");

},

};

const jsonFromObj = **JSON.stringify**(personWithMethod);

*// Метод sayHello был пропущен и не попал в JSON строку*

console.log(jsonFromObj); //{"name":"John","age":26,"isMarried":true}

## JSON.stringify

**JSON.stringify()** - Превращает объект в JSON строку

**// Массив**

const fruitsArray = ["apple", true, "banana", 5, "plum", "cherry", "grape"]; // ['apple', true, 'banana', 5, 'plum', 'cherry', 'grape']

const fruitJSON = JSON.stringify(fruitsArray); // ["apple",true,"banana",5,"plum","cherry","grape"]

//typeof fruitsArray: object

//typeof fruitJSON: string

**// Объект**

const person = {

name: "John",

age: 26,

profession: "Programmer",

isMarried: true,

};

const personJSON = JSON.stringify(person); //{"name":"John","age":26,"profession":"Programmer","isMarried":true}

//main.js:54 typeof personJSON: string

## JSON.parse

**JSON.parse()** - Превращает JSON строку в объект

const string = '["Opel", "Ford", true, "Audi", 5]'; // ["Opel", "Ford", true, "Audi", 5]

const arrayFromString = JSON.parse(string); // ['Opel', 'Ford', true, 'Audi', 5]

**Пример с вложенными объектами**

const person2 = {

name: "John",

age: 26,

profesion: "Programmer",

isMarried: true,

skills: ["JavaScript", "PHP", "HTML", "CSS"],

projects: [

{

projectName: "Online Cinema",

year: 2020,

company: "Yandex",

},

{

projectName: "Pizza Shop",

year: 2020,

company: "Dodo pizza",

},

],

};

const person2JSON = JSON.stringify(person2);

//{"name":"John","age":26,"profesion":"Programmer","isMarried":true,"skills":["JavaScript","PHP","HTML","CSS"],"projects":[{"projectName":"Online Cinema","year":2020,"company":"Yandex"},{"projectName":"Pizza Shop","year":2020,"company":"Dodo pizza"}]}

// typeof person2JSON: string

const person2ObjFromJSON = JSON.parse(person2JSON);

//{name: 'John', age: 26, profesion: 'Programmer', isMarried: true, skills: Array(4), …}age: 26isMarried: truename: "John"profesion: "Programmer"projects: Array(2)0: {projectName: 'Online Cinema', year: 2020, company: 'Yandex'}1: {projectName: 'Pizza Shop', year: 2020, company: 'Dodo pizza'}length: 2[[Prototype]]: Array(0)skills: (4) ['JavaScript', 'PHP', 'HTML', 'CSS'][[Prototype]]: Object

// typeof person2ObjFromJSON: object

## Копирование несложных структур с помощью JSON

JSON можно использовать для глубокого клонирования массивов и объектов

**// Массив**

const fruitsArray1 = ["apple", "banana", "plum", "cherry", "grape"];

const fruitsArray2 = JSON.parse(JSON.stringify(fruitsArray1));

fruitsArray1.push("peach");

console.log(fruitsArray1); //['apple', 'banana', 'plum', 'cherry', 'grape', 'peach']

console.log(fruitsArray2); //['apple', 'banana', 'plum', 'cherry', 'grape']

console.log(fruitsArray1 === fruitsArray2); //false

**// Объект**

const person = {

  name: "John",

  age: 26,

  profesion: "Programmer",

  isMarried: true,

};

const person2 = *JSON*.parse(*JSON*.stringify(person));

delete person.age;

console.log("person: ", person); //{name: 'John', profesion: 'Programmer', isMarried: true}

console.log("person2: ", person2); //{name: 'John', age: 26, profesion: 'Programmer', isMarried: true}

console.log(person === person2); *// false*

Можно также копировать сложные вложенные структуры. Все вложенные структуры также не равны друг другу, копируются.

**!** JSON не кодирует **методы и функции**! (см. пример выше)

**!** JSON не кодирует **рекурсивные структуры**!

const obj = {

a: "Some value",

};

obj.b = obj;

console.log(obj);

//{a: 'Some value', b: {…}}

a: "Some value"

b:

a: "Some value"

b: {a: 'Some value', b: {…}}

[[Prototype]]: Object

[[Prototype]]: Object

JSON.stringify(obj); /Uncaught TypeError: Converting circular structure to JSON