JSON holds data ! (**XML and JSON are both ways of transferring data using HTTP**.) Just like with XML, you put the XML in the body, and if the header Content-Type: application/json, we put JSON in the body.

For example:

Content-Type: application/JSON

“name”: “The Russian” ← we don't have any tags in JSON

The same example in XML:

<name> The Russian </name>

Now we convert previous example with pizza to JSON (смотри xml):

{ //начала описания объекта начинается/заканчивается с фигурных скобок

“pizza”: [ //значение поля состоит из массива в котором тоже лежит 1 обьект

{ //и так как это объект также открываем фигурную скобку и описываем поля

“size” : “Medium”,

“toppings” : [“cheese”, “mushrooms”]

}

{...} //можно добавить ещё обьектов

] //чтобы открыть такой файл в браузере надо накинуть расширение

} //что - то типа json-wue, редактировать с помощью notepad++

//далее нажать **ctrl+O** и выбрать файл на открывание(или указать путь к файлу)

//открыть можно в любом браузере

Comparison XML and JSON:

xml is more powerful, and on the ather hand it makes it more complicated. JSON is very simple that makes it lightweight and smaller. For popularity XML goes down and JSON is going up. When decide wich wone to use it depends on the API. If API is an XML API then we have to provide it XML, and so on with JSON, sometimes theris API with both.

ИЗУЧЕНИЕ JSON

JSON - the way of formatted data. It is basically just a text, (formated) and to parse (разобрать) it and use it in any programming language very easily. It makes data portable from one system to another.

Здесь в фигурных скобках показан объект с полями,описанный в JSON - е каждый объект всегда взят в фигурные скобки, каждое поле взято в кавычки и значение полей также взяты в кавычки.

JSON - is just a key-value way of sending data.

Example:

{ ←обьекты описываются в JSON начиная и заканчивая фигурной скобкой

“id”:1, ← здесь id-это key, а 1 это значение(value), тип данных int он без кавычек

“name”: “The Russian”, ← key (название переменной) это name, а тип данных у поля name - String.

“hobbies” : [“Reading”, “it”, “sport”] ← тип данных-массив of String (массив тоже объект в JS)

}

Ниже пример JSON-а из нескольких объектов, находящихся в array и как и полагается синтаксису в array, объекты разделены comma(запятой). Content отображается во вкладке Pretty

[

{

"id": 1,

"name": "The Russian",

"type": "fiction",

"available": true

},

{

"id": 2,

"name": "Just as I Am",

"type": "non-fiction",

"available": false

},

{

"id": 3,

"name": "The Vanishing Half",

"type": "fiction",

"available": true

},

{

"id": 4,

"name": "The Midnight Library",

"type": "fiction",

"available": true

},

{

"id": 5,

"name": "Untamed",

"type": "non-fiction",

"available": true

},

{

"id": 6,

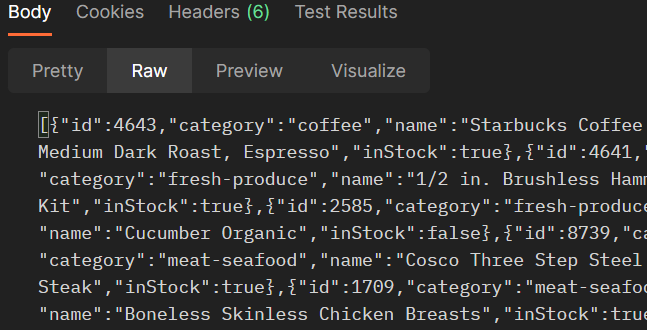
"name": "Viscount Who Loved Me",

"type": "fiction",

"available": true

}

]



А вот так выглядит настоящий JSON который мы получаем, поэтому и вкладка Raw. Raw JSON.

Why do we need to use JSON - we use JSON to transfer data.

For example: we have 2 different systems, and they need to communicate, most of the time they would use JSON to do that. One system may use a programming language, and it is incompatible with the other system. So #1 will use JSON to send data, and the other system that receives it will parse this JSON (transform) into something else that it can process. That is why we use JSON to transfer data.

Spoiler - эту часть я писал до того как учил/разобрался с HTTP

Javascript Object Notation- у объекта в JS (Javascript) есть свойства(поля), с помощью средств JS можно взять этот объект и конвертировать его в String и выслать эту String как сообщение по интернету!!! . API не может выслать тебе на запрос Javascript код, вместо этого оно вышлет JSON, т.е. representation of Javascript object (ту самую String - представление объекта описанного в Javascript, это и есть JSON).

**Ниже код в Javascript**, создаём объект “person” с полями:

let person\_object = { /\* синтаксис создание объекта в JS \*/

name:"Steev",

surname: "Madison",

age: 30,

'city-address': "Mosty", //когда в названии свойства есть дефис, необходимо брать название в одинарные кавычки.

}

let etoJSON\_iz\_obekta\_person\_object=JSON.stringify(person\_object); – **из объекта в JSON**

let object\_from\_JSON=JSON.parse(etoJSON\_iz\_obekta\_person\_object); – **из JSON в объект**

**// Необходимо помнить, что конвертируя объект из JSON-а мы теряем все методы/функции, методы это реализация в javascript, JSON 'заточен' чтобы быть кроссплатформенным средством передачи данных на другие языки программирования и не может в себе нести дополнительную нагрузку реализации методов, только данные.**