JSON на сегодняшний день — общепринятый стандарт обмена данными между сервером и веб-приложением  
Jackson — Java библиoтeka для paбoты c дaнныmи в фopmaтe JSON. Jackson — Oднa из camых быcтpых (выcokoпpoизвoдитeльных) библиoтek

**JSON** (JavaScript Object Notation) - простой формат обмена данными, удобный для чтения и написания как человеком, так и компьютером.

JSON основан на двух структурах данных:

* Коллекция пар ключ/значение. В разных языках, эта концепция реализована как *объект*, запись, структура, словарь, хэш, именованный список или ассоциативный массив.
* Упорядоченный список значений. В большинстве языков это реализовано как *массив*, вектор, список или последовательность.

*Значение* может быть *строкой* в двойных кавычках, *числом*, **true**, **false**, **null**, *объектом* или *массивом*. Эти структуры могут быть вложенными.

JSON имеет следующие преимущества:

* Он компактен.
* Его предложения легко читаются и составляются как человеком, так и компьютером.
* Его легко преобразовать в структуру данных для большинства языков программирования (числа, строки, логические переменные, массивы и так далее)
* Многие языки программирования имеют функции и библиотеки для чтения и создания структур JSON.

Есть несколько основных правил для создания строки JSON:

Строка JSON содержит либо массив значений, либо объект (ассоциативный массив пар имя/значение).

*Массив* заключается в квадратные скобки ([ и ]) и содержит разделенный запятой список значений.

*Объект* заключается в фигурные скобки ({ и }) и содержит разделенный запятой список пар имя/значение.

*Пара имя/значение* состоит из имени поля, заключенного в двойные кавычки, за которым следует двоеточие (:) и значение поля.

*Значение* в массиве или объекте может быть:

Числом (целым или с плавающей точкой)

Строкой (в двойных кавычках)

Логическим значением (true или false)

Другим массивом (заключенным в квадратные скобки)

Другой объект (заключенный в фигурные скобки)

Значение null

<https://ruseller.com/lessons.php?rub=28&id=1212>

Пример

Во многих отношениях вы можете рассматривать JSON как альтернативу XML.преимуществами JSON являются более компактный  и простой для распознавания формат данных.Версию XML также достаточно трудно воспринимать

встроенный метод JSON.stringify(), который берет переменную и возвращает строку JSON, представляющую ее содержание. Метод JSON.stringify() возвращает строку JSON без пробелов. Ее сложнее читать, но зато она более компактна для передачи через сеть.

встроенный метод JSON.parse(). Он получает строку JSON и возвращает объект или массив JavaScript, который содержит данные

<https://medium.com/@stasonmars/%D0%B2%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5-%D0%B2-json-c798d2723107>

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