[{

    "teachers": [{

        "name": "Иванова Ирина Юрьевна",

    "course": "математика",

    "experience": 10

    }, {

        "name": "Юдина Ольга Ивановна",

    "course": "английский",

    "experience": 15

    }]

}, {

    "students": [{

        "name": "Петров Игорь",

    "form": 7,

    "score": 4

    }, {

        "name": "Лесков Олег",

        "form": 10,

        "score": 5

    }]

}, {

    "others": [{

        "name": "Козлова Юлия",

    "position": "директор",

    "department": "управление"

    }, {

        "name": "Козлова Юлия",

    "position": "заведующий учебной частью",

    "department": "управление"

    }]

}]

//new

[{

    "client1": {

        "userId": 1665165165,

        "orderNumber": 15616516,

    "itemsId": [

        15, 178, 451

    ],

    "date": "new Date(154514)"

    }

    }, {

        "client1": {

            "userId": 165165115,

            "orderNumber": 65161611,

        "itemsId": [

            455, 511, 225

        ],

        "date": "new Date(2065665)"

        }

    },{

        "client1": {

            "userId": 665168168,

            "orderNumber": 649856,

        "itemsId": [

            255

        ],

        "date": "new Date(65165651)"

    }

}]

//new

[{

    "patient1": {

        "userId": 1665165165,

        "name": "Петров Глеб Игоревич",

        "theme": 125,

    "phone": "+79771234569",

    "email": "nhh@ya.ru"

    }

    }, {

        "patient2": {

            "userId": 165165115,

            "name": "Афонова Ирина Андреевна",

            "theme": 15,

            "phone": "+79779876541",

            "email": "ppp@ya.ru"

        }

    },{

        "patient3": {

            "userId": 665168168,

            "name": "Королев Алексей Петрович",

            "theme": 22,

            "phone": "+79773215468",

            "email": "yyy@ya.ru"

    }

}]

1. XML – расширяемый язык разметки, который используется для хранения и передачи данных между компьютерными системами. XML документы создаются как инструкции, в соответствии с которыми систематизируются и группируются какие-либо данные для дальнейшего их использования на сайте, внутри приложений компаний, обновления информации с сервера и тд. Такой документ состоит из наборов тегов, из содержимого, атрибутов тегов, которые полностью придумываются разработчиком, затем документ анализируется и спец. программы проверяют связанные с ним файлы на соответствие указанным инструкциям касаемо синтаксиса или значений содержимого. XML документ позволяет отправлять и получать запрос, а XSD проверяет его валидность в целях избегания поломки системы.

XML:

<req>

<field name="dog">dog

</field>

<field name="breed">Beagle</field>

<field name="size">Large</field>

<attribute type="features">

<field name="color">orange</field>

<field name="age">6 years</field>

</attribute>

</req>

JSON:

{"dog": {"breed":"Beagle",

"color":"orange" ,

"size":"Large",

"age": 6}

}

1. Сериализация нужна для конвертирования информации, связанной с объектом, в определенный текстовый формат, который удобен для быстрой передачи или получения каких-либо данных. Она фиксирует данные в определенном состоянии на момент конвертации, которые в дальнейшем с помощью парсинга или десериализации можно обратно распаковать в объект и совершать с ним различные операции.
2. Возможно обработать информацию только в одном виде, текстовом, json, и др.
3. Асинхронные вызовы позволяют обрабатывать запросы на веб-сервере не в одном потоке, а разбивать независимые задачи на разные потоки, которые могут быть выполнены отдельно друг от друга, для того, чтобы при какой-либо ошибке при обработке этих запросов полностью не зависло все приложение и пользователь мог работать с другими задачами, пока часть приложения не работает и не влияет на общий функционал.
4. AJAX позволяет работать с запросами без перезагрузки всей страницы, как если бы мы отправляли данные формы использую Submit, что по умолчанию перезагружает страницу, а асинхронно обновляя только часть информации, связанной с конкретной отправкой запроса.
5. fetch('https://api.github.com/users/Ekaterina-2022')

  .then(response => response.json())

  .then(result => console.log(result))

1. веб-приложение, построенное по принципу spa позволяет создавать сайт, где вся информация расположена на единственной главной странице, при этом при клике на различные вкладки или разделы страница полностью не перезагружается, а лишь подгружает необходимый блок скрипта для этого раздела, который будет отображаться на том же каркасе. что и изначально загруженная страница. По умолчанию главным является код JS, который содержит внутри себя необходимую разметку HTML и стили CSS, поэтому формат такого приложения является упрощенным и обычно создается с помощью фреймворков.