

Формат запросов в REST API

Последние изменения: 2024-03-26

Формат передачи входящих и исходящих данных — JSON, в кодировке UTF-8. Входящие параметры должны передаваться в теле POST запроса или в виде query-строки (?field=value) для GET запросов. Нужно учитывать, что GET запросы имеют ограничение на длину URL — 2048 символов.

В случае, если указан неверный адрес — возвращается ошибка 404.

В случае возникновения ошибки авторизации, возвращается ошибка 401.

Запросы строятся в формате OData, подробнее можно посмотреть по адресу

<http://docs.oasis-open.org/odata/odata/v4.0/odata-v4.0-part2-url-conventions.html>

Пагинация

Функции, возвращающие массив значений, поддерживают параметры для порционной загрузки, которые передаются в виде query-строки (?\$top=1&\$skip=10).

Пример:

[https://localhost:9000/api/v1/Docs?\\$top=5&\\$skip=10](https://localhost:9000/api/v1/Docs?$top=5&$skip=10)

При выполнении данного запроса мы получим максимум 5 записей начиная с 11 индекса в списке.

Если в запросе на получение списка указать параметр count=true, в результате вернется дополнительно поле count, которое содержит общее число записей в списке на сервере.

Фильтрация

Для фильтрации данных в запросе можно указывать параметр \$filter, например:

[https://localhost:9000/api/v1/Docs?\\$filter=lastChangeDate gt 2017-10-06T17:41:10Z and documentTypeName eq 'Заказ'](https://localhost:9000/api/v1/Docs?$filter=lastChangeDate gt 2017-10-06T17:41:10Z and documentTypeName eq 'Заказ')

При выполнении данного запроса мы получим список документов, дата изменения которых более указанной в запросе и тип документа — «Заказ».

Выборка только нужных свойств

Также можно указывать список полей, которые необходимо отобразить в результате, например:

[https://localhost:9005/api/v1/Docs?\\$select=id,tables](https://localhost:9005/api/v1/Docs?$select=id,tables)

Результатом выполнения данного запроса будет список документов, состоящих только из 2-х полей — id и tables.

Выборка записи по идентификатору

Для получения нужной записи необходимо составить запрос с указанием идентификатора этой записи, например:

[https://localhost:9005/api/v1/Docs \('165'\)](https://localhost:9005/api/v1/Docs ('165'))

Сортировка

Чтобы отсортировать результат необходимо указать параметр `$orderby`, например:

[https://localhost:9005/api/v1/Docs?\\$orderby=id](https://localhost:9005/api/v1/Docs?$orderby=id)

В результате получим список документов, отсортированный по идентификатору.

Отображение подтаблиц и строк документов

Для документов реализован вывод записей строк документа и таблиц документа.

Пример:

Если в документе с идентификатором «1» существует таблица «ОплатыВозвраты», то получить строки таблицы можно отправив запрос:

[https://localhost:9000/api/v1/Docs \('1'\)/ОплатыВозвраты](https://localhost:9000/api/v1/Docs ('1')/ОплатыВозвраты)

Для получения только строк документа с идентификатором «123» необходимо выполнить запрос

[https://localhost:9000/api/v1/Docs \('123'\)/declaredItems](https://localhost:9000/api/v1/Docs ('123')/declaredItems) это запрос плановой части документа

[http://localhost:9000/api/v1/Docs \('123'\)?\\$expand=currentItems](http://localhost:9000/api/v1/Docs ('123')?$expand=currentItems) запрос фактической части документа

Развертывание сложных свойств

Чтобы отобразить все документы, включая его таблицы со строками, нужно сформировать запрос в таком виде:

[https://localhost:9000/api/v1/Docs?\\$expand=tables \(\\$expand=rows\)](https://localhost:9000/api/v1/Docs?$expand=tables ($expand=rows))

Поиск без учета регистра

[https://localhost:9000/api/v1/Docs?\\$filter=Contains \(tolower \(name\), 'фраза'\)](https://localhost:9000/api/v1/Docs?$filter=Contains (tolower (name), 'фраза'))

Запросы DELETE

Для запросов DELETE требуется параметр `If-Match`, нужно указывать `*`.

Добавление массива данных

В некоторых методах предусмотрено добавление массива

`{ «value»:[{первый элемент}, {второй элемент}]}`

 интеграция, REST API

Не нашли что искали?



Задать вопрос в техническую поддержку

