Отчет по лабораторной работе №4

Данила Андреевич Стариков

Содержание

# 1 Цель работы

Составить JSON схему, описывающую все возможности метода из лабораторной работы №3 (WEB API Вконтакте). JSON схема должна быть составлена и для объекта, который посылается в запросе и для объекта, который приходит в ответе от сервера. Проверить правильность схемы проверьте с помощью валидатора, например .

# 2 Выполнение лабораторной работы

Будем составлять JSON схему для метода docs.get. Сначала покажем пример работы этого метода:

http POST https://api.vk.com/method/docs.get \  
access\_token==vk1.a.ZdjvFQIhBuM-HEhyBotNorKjYeu\_rjgevnostJtKePu5UuGLz4z8mzOO6GKmrm\_Kk-eA5E8qsKjJbANnt2XU6\_N-Zh5uNZGjHndgbdN\_IftuPuxau-Byr\_jPx9zB5ZO2vXzvTVx27wt5Eg-Da8YH4ZliMHBIOppoSeF2oBd4QedWd0HJ-tBeBRESAQKW22AfdjF1d8kbzaoB2q-CW9Kgeg \  
v==5.199 \  
count==5

Ответ:

{  
 "response": {  
 "count": 329,  
 "items": [  
 {  
 "can\_manage": true,  
 "date": 1702456522,  
 "ext": "pdf",  
 "id": 670812375,  
 "is\_unsafe": 0,  
 "owner\_id": 85081472,  
 "size": 1903102,  
 "title": "Гамильтоновы циклы .pdf",  
 "type": 1,  
 "url": "https://vk.com/doc85081472\_670812375?hash=hS4lZ4ZBWw7Ncbz4ilLTZB2lxIBMZNiSH6UGrONxbGz&dl=a7TsTAQKlhEKhDKYG7HHXzN9pGAKvWoIdJKpVR8VtUX&api=1&no\_preview=1"  
 },  
 {  
 "can\_manage": true,  
 "date": 1678000031,  
 "ext": "pdf",  
 "id": 657272932,  
 "is\_unsafe": 0,  
 "owner\_id": 85081472,  
 "size": 2152747,  
 "title": "Домашнее задание 3.pdf",  
 "type": 1,  
 "url": "https://vk.com/doc85081472\_657272932?hash=dqvgB5Xq26A98155W9ThY3DTJTqXmrLviQsjnoZpfq8&dl=c1CYKNAZ7sjXYzQWuoO3Fbrcdkgm4vsndPhQuKbsJ30&api=1&no\_preview=1"  
 },  
 {  
 "can\_manage": true,  
 "date": 1672062726,  
 "ext": "pdf",  
 "id": 653151850,  
 "is\_unsafe": 0,  
 "owner\_id": 85081472,  
 "size": 4117280,  
 "title": "Подготовка к зачету\_221226\_144325.pdf",  
 "type": 1,  
 "url": "https://vk.com/doc85081472\_653151850?hash=KF5plXD4Gec5C5Xwfcz5zwia0zH5DMUUChWAhHDFxOk&dl=grMOdtdiXgfPJWca8mdDpedQf9CCfs58pbvMk7X4q5o&api=1&no\_preview=1"  
 },  
 {  
 "can\_manage": true,  
 "date": 1671884548,  
 "ext": "pdf",  
 "id": 653011190,  
 "is\_unsafe": 0,  
 "owner\_id": 85081472,  
 "size": 2515804,  
 "title": "Notes\_221224\_152122.pdf",  
 "type": 1,  
 "url": "https://vk.com/doc85081472\_653011190?hash=ZuzwpsckqWrEOXipCh2ep0b9lrNjry38pYxBZ2jYriH&dl=qfTk5Mjex9j3tOJYwKCkQOGVtVvZqPN0vMZTFxKFh8P&api=1&no\_preview=1"  
 },  
 {  
 "can\_manage": true,  
 "date": 1671535324,  
 "ext": "pdf",  
 "id": 652721733,  
 "is\_unsafe": 0,  
 "owner\_id": 85081472,  
 "size": 8594963,  
 "title": "Подготовка к кр\_221207\_082207.pdf",  
 "type": 1,  
 "url": "https://vk.com/doc85081472\_652721733?hash=Seblnf8He5kCbmEhTGZOj2r51IoGs9mYhIR3UgzRV10&dl=h6LDxvWYnUoogJxsQ3CwF9IqvFgzkKLVX8Q0znhJMz4&api=1&no\_preview=1"  
 }  
 ]  
 }  
}

Теперь запишем JSON схему для запроса к этому методу:

{  
 "name": "docs.get",  
 "description": "Возвращает расширенную информацию о документах пользователя или сообщества.",  
 "access\_token\_type": [  
 "user"  
 ],  
 "parameters": [  
 {  
 "name": "count",  
 "description": "Количество документов, информацию о которых нужно вернуть. По умолчанию: все документы. Максимальное количество документов, которое можно получить: 2000.",  
 "type": "integer",  
 "maxItems": 2000,  
 "minimum": 1  
 },  
 {  
 "name": "offset",  
 "description": "Смещение, необходимое для выборки определенного подмножества документов. Максимальное значение: 1999",  
 "type": "integer",  
 "maxItems": 1999,  
 "minimum": 1  
 },  
 {  
 "name": "type",  
 "description": "Тип файла. Возможные значения:",  
 "type": "integer",  
 "minimum": 1,  
 "maximum": 8  
 },  
 {  
 "name": "owner\_id",  
 "description": "Идентификатор пользователя или сообщества, которому принадлежат документы.",  
 "type": "integer"  
 },  
 {  
 "name": "return\_tags",  
 "description": "",  
 "type": "integer",  
 "minimum": 0,  
 "maximum": 1  
 },  
 ],  
 "responses": {  
 "response": {  
 "$ref": "#/docs\_get\_response"  
 }  
 }  
}

Опишем структуру объекта, приходящего в ответ:

{  
 "docs\_get\_response": {  
 "type": "object",  
 "properties": {  
 "response": {  
 "type": "object",  
 "properties": {  
 "count": {  
 "type": "integer",  
 "description": "Количество элементов в массиве \"items\"",  
 "minimum": 0  
 },  
 "items": {  
 "type": "array",  
 "description": "Массив объектов файлов",  
 "items": {  
 "$ref": "#/definitions/object\_file"  
 }  
 }  
 }  
 }  
 },  
 "required": [  
 "response"  
 ],  
 "additionalProperties": false  
 }  
}

Запишем JSON схему для объекта файла:

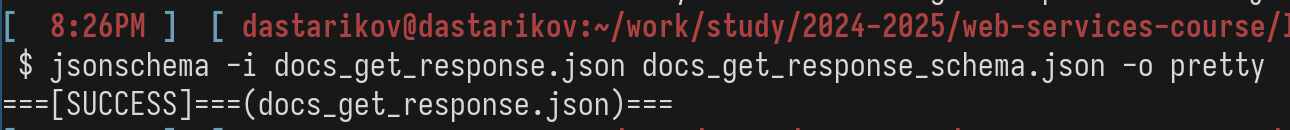
{  
 "object\_file":{  
 "type": "object",  
 "description": "Объект, описывающий файл",  
 "properties": {  
 "id": {  
 "type": "integer",  
 "description": "Идентификатор файла"  
 },  
 "owner\_id": {  
 "type": "integer",  
 "description": "Идентификатор пользователя, загрузившего файл"  
 },  
 "title": {  
 "type": "string",  
 "description": "Название файла"  
 },  
 "size": {  
 "type": "integer",  
 "description": "Размер файла в байтах"  
 },  
 "ext": {  
 "type": "string",  
 "description": "Расширение файла"  
 },  
 "url": {  
 "type": "string",  
 "description": "Адрес файла, по которому его можно загрузить"  
 },  
 "date": {  
 "type": "integer",  
 "description": "Дата добавления в формате Unixtime"  
 },  
 "type": {  
 "type": "integer",  
 "description": "Тип файла. Возможные значения:1 — текстовые документы; 2 — архивы; 3 — gif; 4 — изображения; 5 — аудио; 6 — видео; 7 — электронные книги; 8 — неизвестно.",  
 "minimum": 1,  
 "maximum": 8  
 },  
 "preview": {  
 "type": "object",  
 "description": "Информация для предварительного просмотра файла.",  
 "properties": {  
 "photo": {  
 "type": "object",  
 "properties": {  
 "sizes": {  
 "type": "array",  
 "items": {  
 "type": "object",  
 "properties": {  
 "src": {  
 "type": "string"  
 },  
 "width": {  
 "type": "integer",  
 "mininum": 0  
 },  
 "height": {  
 "type": "integer",  
 "mininum": 0  
 },  
 "type": {  
 "enum": ["s", "m", "x", "o", "p", "q", "r", "y", "z", "w"]  
 }  
 }  
 }  
 }  
 }  
 },  
 "graffiti": {  
 "type": "object",  
 "properties": {  
 "src": {  
 "type": "string"  
 },  
 "width": {  
 "type": "integer",  
 "minimum": 0  
 },  
 "height": {  
 "type": "integer",  
 "minimum": 0  
 }  
 }  
 },  
 "audio\_message": {  
 "type": "object",  
 "properties": {  
 "duration": {  
 "type": "integer"  
 },  
 "waveform": {  
 "type": "array",  
 "items": {  
 "type": "integer"  
 }  
 },  
 "link\_ogg": {  
 "type": "string"  
 },  
 "link\_mp3": {  
 "type": "string"  
 }  
 }  
 }  
 }  
 }  
 }  
 }  
}

Для проверки созданной JSON схемы создали два файла: docs\_get\_response\_schema.json, где описана JSON схема, и docs\_get\_response.json, куда записан ответ, пришедший при выполнении следующего запроса:

http POST https://api.vk.com/method/docs.get \  
access\_token==vk1.a.OCl64Tbc6nXOzplPxmFW0iK6OHlN08C2QmttWJDBRbUzGn7BIfs6n50Tcp59JvBtsGNKLGv4z1hWXX8NUgEwQ6p9Vt7UIu6p4LjOx561kybTwVXPMuFjtHdX6Is\_EiqBZB1Z8tdfFns8TKix0xsvzYDyamJWVSeWLyOsqWvQ9cN59H5V6hbQ7Ktp1UYev5FY6X-60XLwjV1qi7sF0HwJYw \  
count==25 \  
v==5.199 \  
-o docs\_get\_response.json

Далее с помощью пакета [jsonschema](https://pypi.org/project/jsonschema/) в командной строке запустили проверку:

jsonschema -i docs\_get\_response.json docs\_get\_response\_schema.json -o pretty



Результат проверки ответа сервера с помощью созданной JSON схемы

# 3 Выводы

В ходе выполнения лабораторной работы разработали JSON схему по стандарту JSON Schema на основе метода docs.get API Вконтакте.