

ВОЗМОЖНОСТИ ЯЗЫКА ПРОГРАММИРОВАНИЯ VKSCRIPT

FEATURES OF THE VKSCRIPT PROGRAMMING LANGUAGE

A. Pimanov

Summary: The VKontakte social network has its own sophisticated tool for interacting with its components. The data of the tools is an API consisting of methods. Among the many methods, there is one non-standard one that is used to execute code written in the VKScript programming language.

VKScript was developed as a proprietary script language for the VKontakte social network, with the aim of creating custom applications and facilitating the automation of everyday tasks.

This article discusses the capabilities and features of the VKScript programming language, how to work with it, and also provides examples of its use.

Keywords: VKScript, API, JSON, JavaScript, VKontakte, method.

VKScript — это язык программирования, разработанный для социальной сети ВКонтакте, который позволяет написать скрипты для выполнения определенных задач на платформе ВКонтакте. VKScript базируется на JavaScript или ActionScript и использует специальные функции и методы, которые позволяют взаимодействовать с API ВКонтакте.

Так как синтаксис языка базируется на JavaScript, рассмотрим первую программу любого разработчика на этом языке.

Листинг 1

```
alert(«Hello world!»);
```

Для того чтобы выполнить данную программу, необходимо отправить JSON запрос используя метод API ВКонтакте — `execute`. Можно запустить описанную программу ещё одним способом перейдя на страницу документации данного метода, где в разделе «Пример запроса» метода можно выполнить VKScript код. Данный раздел используется для того, чтобы продемонстрировать работу любого метода API ВКонтакте посредством отправки JSON запроса. Метод `execute` для версии API 5.131 требует 4 параметра. Перечень требуемых параметров приведён в таблице:

Параметр	Описание
<code>access_token</code>	Специальный ключ доступа, используемый для верификации. Он представляет собой строку из латинских букв и цифр и может соответствовать отдельному пользователю, сообществу или приложению.

Пиманов Андрей Евгеньевич

Северо-Восточный федеральный университет
имени М.К. Аммосова, г. Норильск
epimanov15@gmail.com

Аннотация. Социальная сеть ВКонтакте имеет свой сложноустроенный инструмент для взаимодействия с её составляющими. Данным инструментом является API, состоящий из методов. Среди множества методов присутствует один нестандартный, который используется для выполнения кода написанного на языке программирования VKScript.

VKScript был разработан как собственный язык скриптов для социальной сети ВКонтакте, с целью создания пользовательских приложений и облегчения автоматизации повседневных задач.

В данной статье рассмотрены возможности и особенности языка программирования VKScript, способы работы с ним, а также приведены примеры его использования.

Ключевые слова: VKScript, API, JSON, JavaScript, ВКонтакте, метод.

Параметр	Описание
<code>code</code>	Код алгоритма на языке программирования VKScript.
<code>v</code>	Версия API ВКонтакте.

Попытка выполнить данный код приведёт к следующей ошибке — *Unable to compile code: undefined identifier 'alert' in line 1*. Замена `alert` другими операторами вывода не приведёт к устранению ошибки. Следовательно, VKScript не поддерживает операторы промежуточного вывода, что абсолютно логично, так как у них нет практического применения с точки зрения области применения самого языка программирования. Но фразу «Hello world!» вывести всё же возможно. Для этого необходимо вернуть результат выполнения программы как это делается для функций.

Листинг 2

```
return(«Hello world!»);
```

Важно понимать, что оператор `return` завершает применение программы. Результатом выполнения приведённого ниже кода будет вывод фразы — `first output`.

Листинг 3

```
return(“first output”);  
return(“second output”);
```

Во многих аспектах устройство языка схоже с JavaScript или ActionScript, однако стоит учитывать, что VKScript имеет множество ограничений. VKScript поддерживает: логические операции; арифметические опе-

рации; условные операторы; циклы; массивы и списки; некоторые методы JavaScript. Пример цикла, реализованного на языке VKScript:

Листинг 4

```
var a = 1;
var b = 10;
while (b != 0) {
    b = b - 1;
    a = a + 1;
};
return a;
```

В данном коде объявляются и инициализируются переменные **a** и **b** для демонстрации работы цикла. Пока **b** не будет равно 0, значение переменной **b** будет уменьшаться на единицу, а значение переменной **a** будет увеличиваться на единицу. Когда цикл завершится, программа вернет значение переменной **a**, которое будет равно 11, и выполнение программы прекратится.

Главной особенностью VKScript является возможность вызова других методов API ВКонтакте. Таким образом, предоставляется широкий спектр функций для работы с ВКонтакте, включая доступ к информации о профилях, группах, фотографиях, сообщениях и так далее. Для того чтобы вызвать метод ВКонтакте необходимо использовать следующую конструкцию — API.имя_метода. В следующем примере кода продемонстрирована работа с методами ВКонтакте для установки и получения статуса. Данные методы носят названия status.get и status.set. У каждого из них есть свои обязательные параметры. Для метода status.get это:

Параметр	Описание
access_token	Специальный ключ доступа, используемый для верификации.
text	Текст нового статуса.
group_id	Идентификатор сообщества, в котором будет установлен статус. По умолчанию статус устанавливается текущему пользователю.
v	Версия API ВКонтакте.

Для метода status.set это:

Параметр	Описание
access_token	Специальный ключ доступа, используемый для верификации.
user_id	Идентификатор пользователя или сообщества, информацию о статусе которого нужно получить.
group_id	Идентификатор сообщества, статус которого необходимо получить.
v	Версия API ВКонтакте.

Листинг 5

```
var data = {«text»: «Тестовый статус»};
API.status.set(data);
return(API.status.get());
```

В качестве результата выполнения будет возвращён новый установленный на странице пользователя статус — Тестовый статус. Также важно учитывать, что VKScript чувствителен к регистру и изменение регистра в наименовании статуса приведёт к фатальной ошибке.

До этого момента было не очень понятно, какие явные преимущества даёт VKScript, ведь все операции, которые возможно выполнить с использованием данного языка программирования можно повторить посредством любого другого языка программирования. Преимущество использования VKScript совсем не очевидно и заключается в возможности преодолеть лимит на количество запросов. Данное преимущество кажется совсем не значительным, однако рассматривая устройство API ВКонтакте можно убедиться в обратном. Например, рассмотрим ситуацию, в которой необходимо получить список сообщений от лица сообщества в беседе, в которую оно было добавлено. Для этого можно воспользоваться любым из методов, возвращающих историю переписки, но не один из этих методов не вернёт её. Получить необходимый результат можно только в двух случаях: беседа создана сообществом, от имени которого будет выполняться запрос; история сообщений запрашивается из личных сообщений. Вероятно, это связано с обеспечением конфиденциальности пользователей социальной сети. Техническая поддержка ВКонтакте предлагает следующее решение проблемы. Чтобы получить текст сообщения можно использовать метод messages.getByConversationMessageId. Данный метод возвращает сообщения по их идентификатору (conversation_message_id) и принимает следующие параметры:

Параметр	Описание
access_token	Специальный ключ доступа, используемый для верификации.
peer_id	Идентификатор чата.
conversation_message_ids	Идентификаторы сообщений. Максимум 100 сообщений.
extended	Определяет необходимость получения в ответе дополнительных полей. Принимает логический тип данных.
fields	Список запрашиваемых дополнительных полей в ответе.
group_id	Идентификатор сообщества (для сообщений сообщества с ключом доступа пользователя).
v	Версия API ВКонтакте.

Данным методом получить историю достаточно длинной переписки не удастся, используя любой другой язык программирования. Причиной этому станет ограничение на количество выполняемых запросов за раз. Обращаясь к методу `messages.getByConversationMessageId` посредством языка программирования `VKScript`, данное ограничение можно обойти. В листинге 6 приведена возможная реализация данной задачи.

Листинг 6

```
// Список идентификаторов сообщений, которые необходимо получить
var conversation_message_ids = [1, 2, 3];
var peer_id = идентификатор_чата;

var data = [];

var i = 0;
while (i < conversation_message_ids.length) {
  var request = API.messages.getByConversationMessageId({"peer_id": peer_id, "conversation_message_ids": conversation_message_ids[i]});
  data.push([conversation_message_ids[i], request.items[0].text]);
  i = i + 1;
}

return(data);
```

Данный код вернёт JSON с двумерным массивом, где в каждый вложенный массив будет вложен идентификатор сообщения и его текст.

Несмотря на то, что `VKScript` дает возможность обойти ограничения на количество запросов, для самого языка установлены ограничения на их выполняемое количество. Увеличив количество элементов в массиве `conversation_message_ids` из листинга 6, можно установить, что количество допустимых вызовов равно 25. За пределами этого значения можно получить ошибку — `Runtime error occurred during code invocation: Too many API calls` (Во время вызова кода произошла ошибка времени выполнения: слишком много вызовов API). Следует выяснить, существует ли разница в допустимом количестве вызовов между методами и определить, является ли ограничение на количество вызовов индивидуальным для каждого метода при выполнении `VKScript` кода. С помощью кода в листинге 7 проверим первую гипотезу.

Листинг 7

```
var data = [];
var count = 25;
var i = 0;
```

```
while (i < count) {
  data.push(API.status.get());
  i = i + 1;
}

return(data);
```

Данный код возвращает JSON массив из одинаковых статусов, взятых из одного источника. Запрос будет выполняться 25 раз, однако увеличив значение переменной `count` на единицу, скрипт начнёт завершаться фатальной ошибкой, описанной ранее (ограничивающей количество запросов). Это говорит о том, что каждый запрос можно выполнить одинаковое количество раз. Метод `status.get` был выбран, так как он и метод `messages.getByConversationMessageId` располагаются в разных разделах API ВКтакте. Так же заменяя метод `status.get` на любой другой, максимальное количество запросов, которое можно будет выполнить новым методом, останется равным 25. Чтобы проверить следующую гипотезу приводится листинг 8.

Листинг 8

```
// Список идентификаторов сообщений, которые необходимо получить
var conversation_message_ids = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25];
var peer_id = идентификатор_чата;

var data = [];

var i = 0;
while (i < conversation_message_ids.length) {
  var request_first = API.messages.getByConversationMessageId({"peer_id": peer_id, "conversation_message_ids": conversation_message_ids[i]});
  var request_second = API.status.get();
  data.push([conversation_message_ids[i], request_first.items[0].text, request_second]);

  i = i + 1;
}

return(data);
```

Данный код дублирует код из листинга 6, с тем отличием, что в него включён метод `status.get`. Информация, полученная данным методом вложена в ранее описанный JSON и возвращена после завершения работы цикла `while`. Выполнение данной программы приведёт к ошибке, ограничивающей максимальное количество выполняемых за раз запросов. Снизив количество обрабатываемых идентификаторов до 13, проблема не решится, так как суммарное количество выполненных запросов будет равно 26. Снизив количество обрабатываемых иденти-

фикаторов до 12, количество выполненных запросов составит 24, что уже не приведёт к фатальной ошибке и результатом программы станет оговорённый JSON. Из этого следует, что ограничение в 25 запросов выделяется на один вызов метода `execute`.

Выявленные аспекты языка программирования VKScript говорят о том, что его сильная сторона не настолько существенна насколько может показаться, особенно учитывая тот факт, что VKScript хоть и схож с JavaScript и ActionScript, но сам по себе очень сильно ограничен и не поддерживает множество операторов и методов языков прототипов. Из этого следует вывод: VKScript не стоит рассматривать как самостоятельное решение и пользу из его включения в проект можно получить только посредством взаимодействия с любым другим языком программирования. Например, в рамках условно взятого PHP формируется строка с VKScript кодом и отдаётся на выполнения методу `execute`. Рассматривая листинг 6, можно формировать массив `conversation_message_ids` на 25 элементов из строк содержащих идентификаторы сообщений разделённый через запятую. То есть первый элемент представляет из себя строку «1, 2, 3, ..., 100», а второй «101, 102, 103, ..., 200». Таким образом, можно передать до 2500 идентификаторов сообщений, так как параметр `conversation_message_ids` метода `messages.getByConversationMessageId` способен принимать максимум 100 идентификаторов. Следова-

тельно, за выполнение одного метода `execute` можно получить до 2500 сообщений. Например, если за раз можно было бы обратиться к API ВКонтакте используя любой другой язык программирования только 10 раз, то обращаясь к методу `messages.getByConversationMessageId` без использования VKScript получилось бы извлечь до 1000 сообщений. Но обращаясь к методу `execute` передавая ему VKScript код, можно было бы извлечь до 25000 сообщений. Именно такой вариант использования VKScript в связке с любым другим языком программирования демонстрирует его сильную сторону и делает это преимущество более существенным.

Данный материал охватил описание синтаксиса и структуры языка программирования VKScript, созданного ВКонтакте для решения задач, связанных с работой с API. Так же были рассмотрены способы работы с ним. Была выполнена аналитическая работа, выявившая следующие недостатки: ограниченность набора функций и операторов; низкое количество выполняемых запросов к API за один вызов метода `execute`; ограниченная документация. Кроме найденных недостатков, был выявлен оптимальный способ работы с методом `execute`, принимающим VKScript код. Это позволяет выполнять существенно больше запросов к API ВКонтакте, используя всего один запрос к соответствующему методу. Таким образом, VKScript представляет собой ценный инструмент для автоматизации своих задач на платформе.

ЛИТЕРАТУРА

1. Официальный веб-сайт «ВКонтакте», раздел «Для разработчиков» [Электронный ресурс]. URL: <https://dev.vk.com/reference> (дата обращения: 29.04.2023).
2. Веб-сайт «Хабр», статья «Анализ языка VKScript: JavaScript, ты ли это?» [Электронный ресурс]. URL: <https://habr.com/ru/articles/464099/> (дата обращения: 29.04.2023).
3. Пиманов, А.Е. Выбор средств разработки чат-бота для игры в монополию «в контакте» / А.Е. Пиманов // Материалы XXII Всероссийской научно-практической конференции молодых ученых, аспирантов и студентов, с международным участием в г. Нерюнгри, посвященной 30-летию юбилею Технического института (филиала) СВФУ им. М.К. Аммосова : Материалы конференции, Нерюнгри, 28–29 октября 2022 года / Редколлегия: А.В. Рукочин, Н.Н. Гриб, П.Ю. Кузнецов [и др.]. — Якутск: Северо-Восточный федеральный университет имени М.К. Аммосова, 2022. — С. 247–250. — DOI 10.52994/9785751333737_058. — EDN UHWRXQ.
4. Лихач, А.А. Чат-бот ВКонтакте расписания учебных занятий университета / А.А. Лихач, П.В. Татаренко // Вестник Новгородского государственного университета. — 2022. — № 3(128). — С. 120–125. — DOI 10.34680/2076-8052.2022.3(128).120–125. — EDN TSYFLN.
5. Дейнеко, Т.А. Чат-бот ВКонтакте «Расписание занятий ОмГУ» / Т. Дейнеко, Д. Бобров // Математические структуры и моделирование. — 2020. — № 3(55). — С. 117–122. — DOI 10.24147/2222-8772.2020.3.117–122. — EDN RJQPCL.

© Пиманов Андрей Евгеньевич (epimanov15@gmail.com)

Журнал «Современная наука: актуальные проблемы теории и практики»