

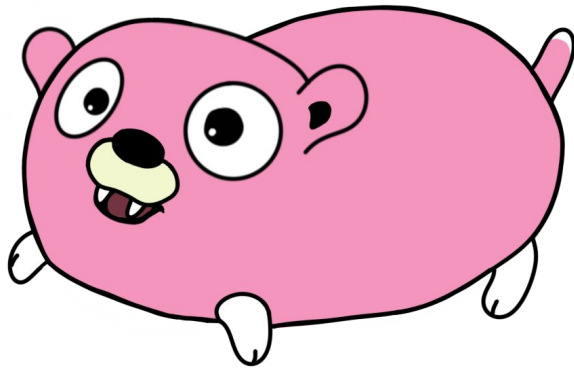


Используем VK API из Go



0 ce6e

- ❑ Web apps, full stack (PHP, Go, JS)
- ❑ Database firewall (C++)
- ❑ Go compiler (Go + x86 asm) @ Intel
- ❑ Backend infrastructure (Go) @ VK



quasilyte.dev/status

Зачем мы собрались?

Хорошо провести время и узнать что-то новое



Структура семинара

- Скачайте эти слайды!



vk_api_from_go

Структура семинара

- Скачайте эти слайды!
- Подготовка и проверка окружения



vk_api_from_go

Структура семинара

- ❑ Скачайте эти слайды!
- ❑ Подготовка и проверка окружения
- ❑ Краткий обзор VKontakte API



vk_api_from_go

Структура семинара

- ❑ Скачайте эти слайды!
- ❑ Подготовка и проверка окружения
- ❑ Краткий обзор VKontakte API
- ❑ Настраиваем доступ к API



vk_api_from_go

Структура семинара

- ❑ Скачайте эти слайды!
- ❑ Подготовка и проверка окружения
- ❑ Краткий обзор VKontakte API
- ❑ Настраиваем доступ к API
- ❑ Рассматриваем демо-приложение



vk_api_from_go

Структура семинара

- ❑ Скачайте эти слайды!
- ❑ Подготовка и проверка окружения
- ❑ Краткий обзор VKontakte API
- ❑ Настраиваем доступ к API
- ❑ Рассматриваем демо-приложение
- ❑ Делаем своё приложение



vk_api_from_go

Если нет ноутбука

Найдите себе напарника с ноутбуком



Подготовка окружения

1. Установка Go
2. Hello, World!
3. GOPATH и модули
4. Пара слов об IDE/редакторах для Go

Let's do this!



Для опытных

Пока остальные устанавливают и настраивают Go, вы можете:

- ❑ изучить [документацию API](#)
- ❑ помочь остальным с настройкой Go
- ❑ придумать, какое приложение вы будете делать

Устанавливаем Go

1. Скачать дистрибутив для своей OS
2. Далее действия зависят от OS

На Windows Go будет установлен в C:\Go, а C:\Go\bin будет добавлен в системный PATH.

Для macOS тоже есть инсталлятор.

<https://golang.org/dl/>

https://golang.org/dl/



Search

The Go Programming Language

Documents

Packages

The Project

Help

Blog

Search



Downloads

After downloading a binary release suitable for your system, please follow the [installation instructions](#).

If you are building from source, follow the [source installation instructions](#).

See the [release history](#) for more information about Go releases.

Featured downloads

Microsoft Windows

Windows 7 or later, Intel 64-bit processor

[go1.12.4.windows-amd64.msi](#) (118MB)

Apple macOS

macOS 10.10 or later, Intel 64-bit processor

[go1.12.4.darwin-amd64.pkg](#) (121MB)

Linux

Linux 2.6.23 or later, Intel 64-bit processor

[go1.12.4.linux-amd64.tar.gz](#) (122MB)

Source

[go1.12.4.src.tar.gz](#) (21MB)

Для Windows

1. Запустить инсталлятор.

Здесь и далее считаем, что после этого у вас доступна команда “go” в терминале (cmd.exe).

Для Linux

1. Распаковать архив в `/usr/local/go`
2. Добавить `/usr/local/go/bin` в системный PATH

Hello, World! (1)

Создайте файл "hello.go":



```
package main

func main() {
    println("Golang + GDG Nizhny")
}
```

Hello, World! (2)



Запустите hello.go через терминал:

```
$ go run hello.go  
Golang + GDG Nizhny
```

GORATH и модули

- ❑ GORATH - переменная окружения, которая указывает где Go будет искать импортируемые пакеты.

То, что находится внутри GORATH именуют “Go workspace”. Ближе всего он парадигме моно-репозиториев.

GORATH и модули

- ❑ Go modules - будущее управления зависимостями в Go. Сделает GORATH избыточным.

Что использовать - выбор за вами.

Сегодня вам помогут с любым из вариантов.

(Предпочтительно брать модули, но в проектах из одного файла разницы нет никакой.)

В чём писать код на Go?

Основные варианты:

- ▢ Visual Studio Code + Go plugin
- ▢ Goland (IntelliJ IDE)

Остальные варианты (типа Emacs/Vim/Sublime Text) тоже работают, но в них больше подводных камней.

Краткий обзор VKontakte API



<https://vk.com/dev/manuals>

Документация и управление приложениями



Об API

- ❏ Доступ по HTTP
- ❏
- ❏
- ❏

Использовать API можно хоть через curl.
Но SDK упрощает жизнь разработчиков.

Об API

- ❑ Доступ по HTTP
- ❑ Операции (методы) делятся по категориям
- ❑
- ❑

Категориями являются пользователи (user), стены (wall), сообщества (group) и [прочие](#).

Об API

- ❑ Доступ по HTTP
- ❑ Операции (методы) делятся по категориям
- ❑ Разграничение прав (scope) по категориям
- ❑

Используется API на определённом scope. Например, токен может быть выдан только на доступ к друзьям.

Об API

- ❑ Доступ по HTTP
- ❑ Операции (методы) делятся по категориям
- ❑ Разграничение прав (scope) по категориям
- ❑ Возвращаемые ответы в формате JSON

Все ответы сервера приходят в JSON.

У API есть [схема](#).

VK SDK

- ❑ Java, JS, Flash, Android, iOS, PHP - [официальные](#)
- ❑ <https://github.com/Vorkytaka/easyvk-go> ~2018
- ❑ <https://github.com/urShadow/go-vk-api> ~2017

Официальных SDK для Go нет, но есть разные наработки в Open Source. Мы их использовать не будем, но вы можете изучить их для интереса.

Что можно сделать через VK API?

- ❑ Извлекать информацию: людей, посты, сообщения
- ❑ Создавать контент
- ❑ Управлять группой или своей страницей
- ❑ Отслеживать события (и реагировать на них)
- ❑ Хранить состояние в [storage](#) (get/set)

Посмотрите [список методов](#),
он довольно внушителен.

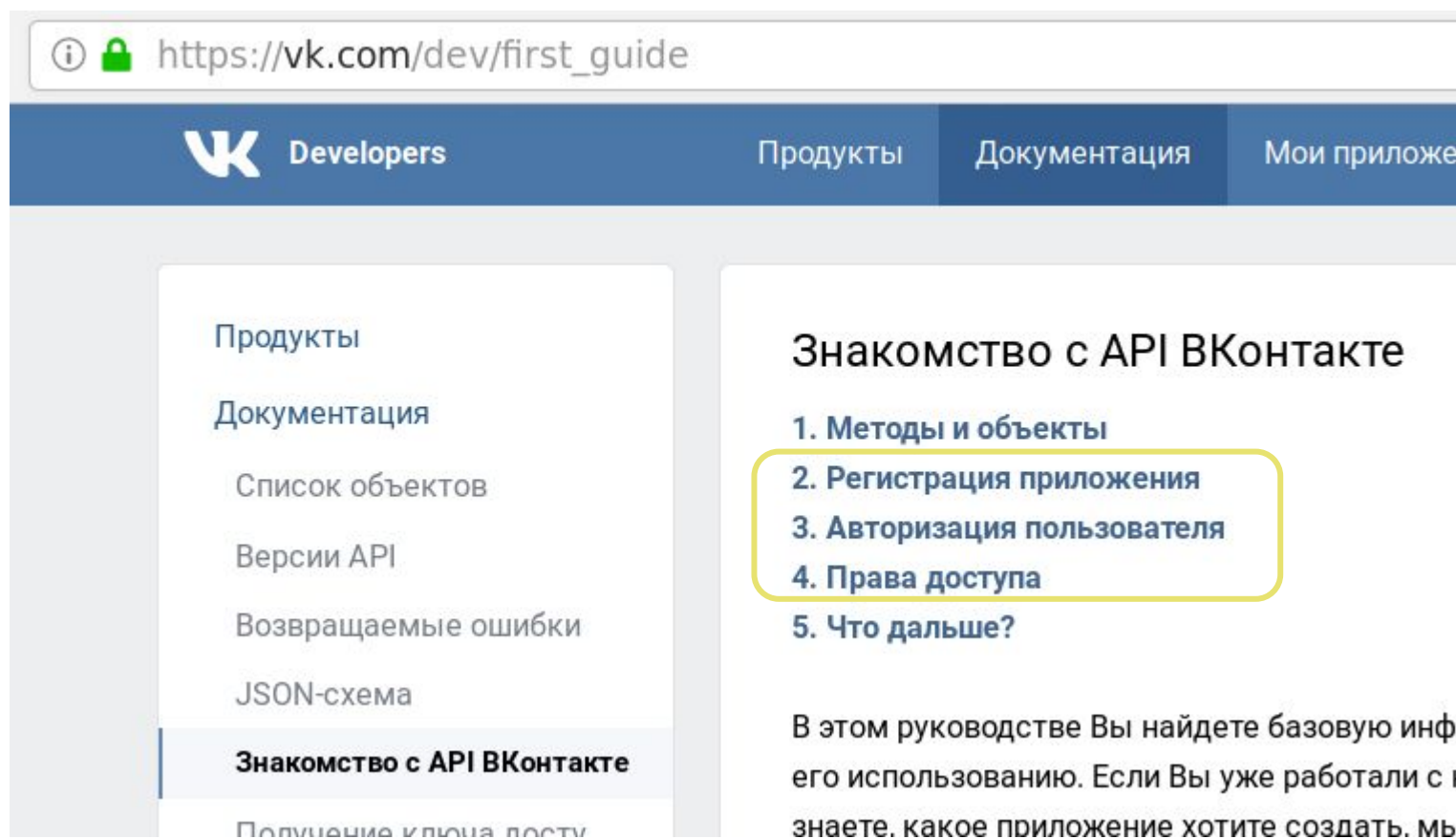
Настраиваем доступ к API






Вам потребуется аккаунт в ВК
Если у вас его нет, найдите себе команду



https://vk.com/dev/first_guide



<https://vk.com/editapp?act=create>

📄  https://vk.com/editapp?act=create ⋮  

VK Developers Продукты Документация Мои приложения Поддержка

Условия размещения при...

Продукты

Документация

SDK

Поддержка

Список методов

Создание приложения

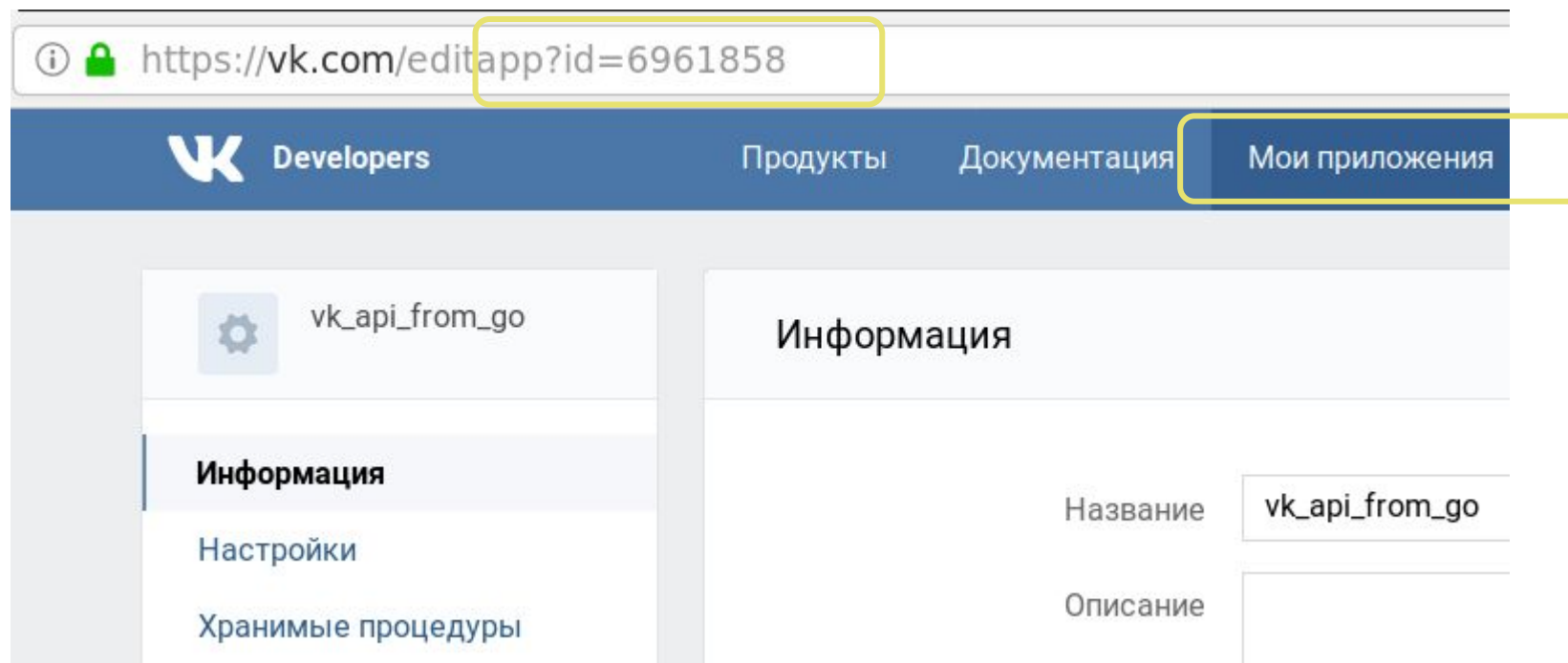
Название

Платформа: ☐ Встраиваемое приложение

☒ Standalone-приложение

☐ Веб-сайт

Скопируйте ID приложения



Получение access_token (A)

Откройте URL в браузере:

Замените **ID** идентификатором приложения.
Токен выдаётся на сутки.

https://oauth.vk.com/authorize?client_id=ID&display=page&scope=friends&response_type=token&v=5.52

Получение access_token (B)

Откройте URL в браузере:

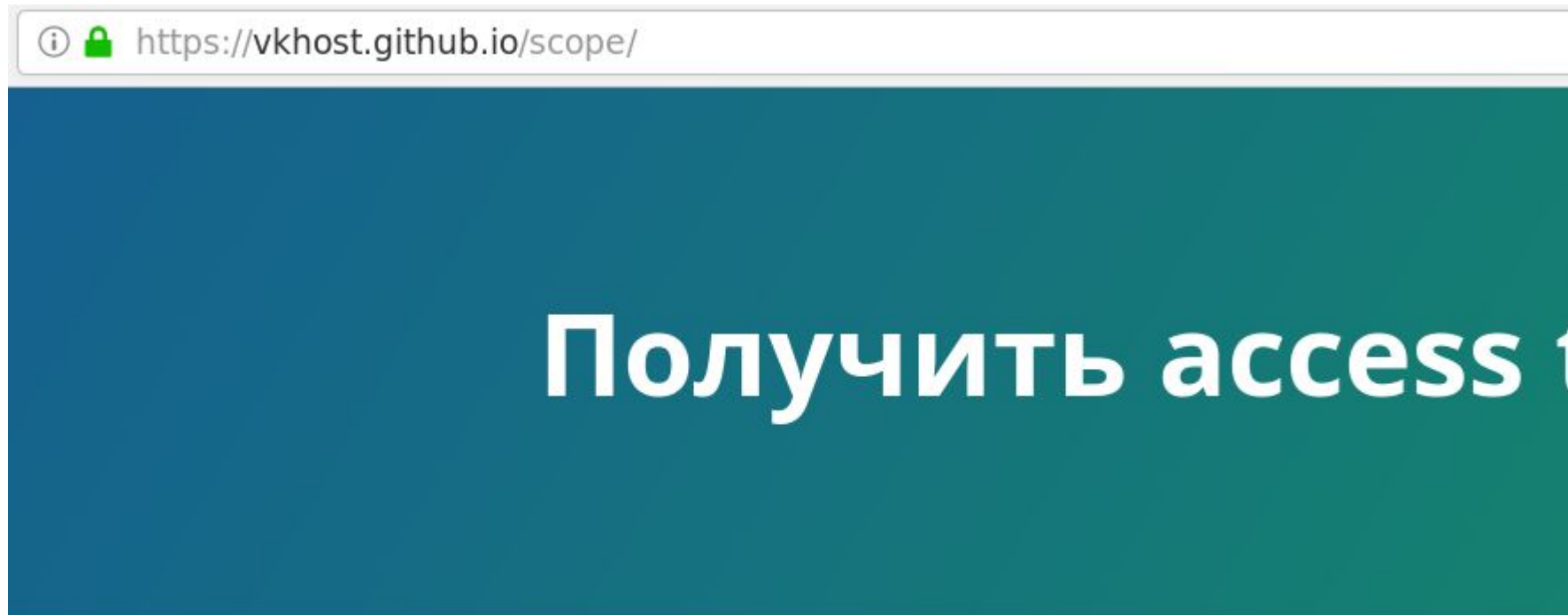
<https://vkhost.github.io/scope/>

Введите:

- friends (scope доступа)
- ID вашего приложения

(См. скриншот на следующем слайде.)

<https://vkhost.github.io/scope/>



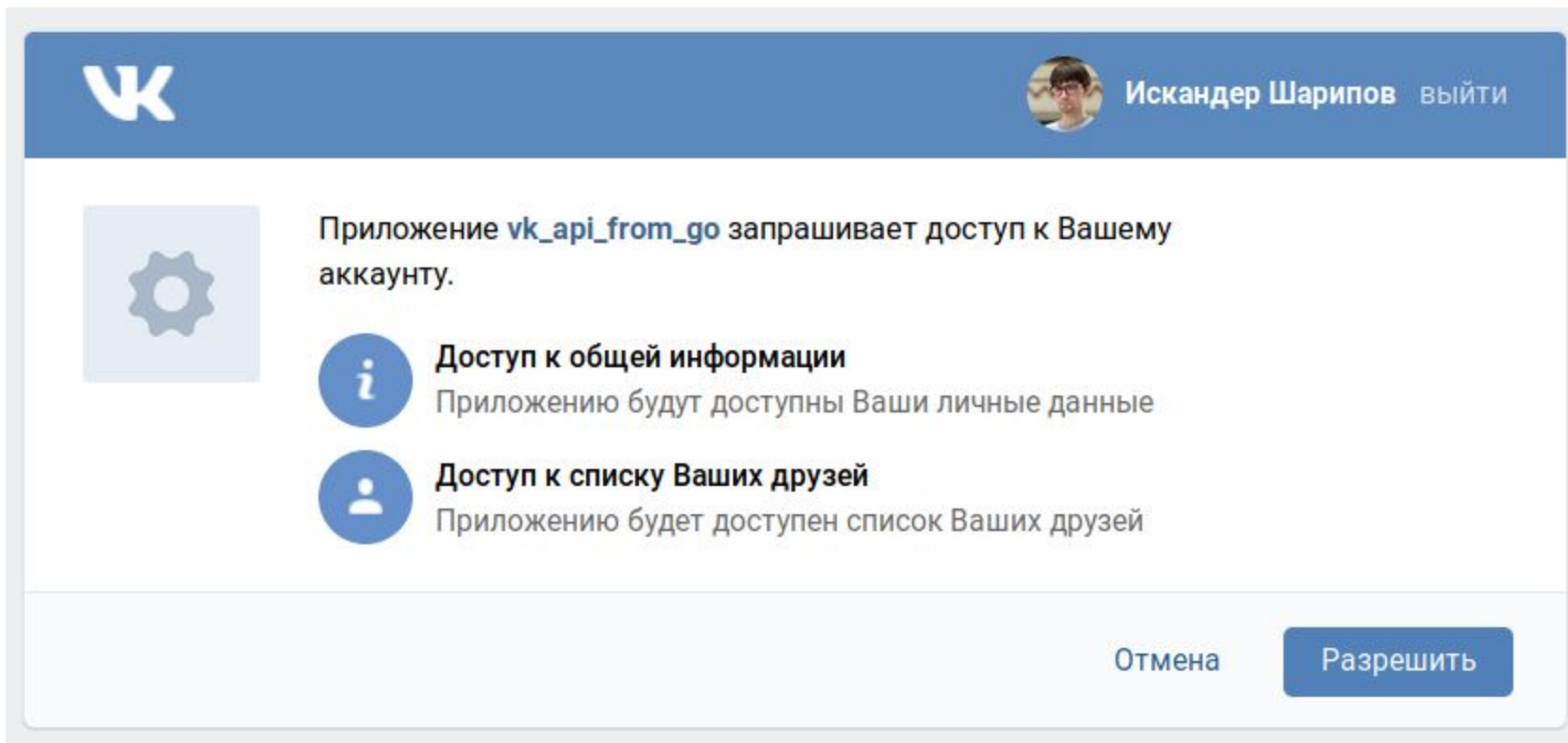
Введите права доступа через запятую: friends

Введите ID приложения:

Готово

Внимание! Не убирайте право "offline", если планируете использовать токен с друг

Разрешите доступ



Скопируйте из URL access_token

  https://oauth.vk.com/blank.html#access_token=dcff7c8946ce2a10066dc015bdc2d5   

Ограничения

На токен, выданный пользователю, накладываются некоторые ограничения:

- ❑ Частотность: не более 3-х запросов в секунду
- ❑ Количество: имеется спам-защита
- ❑ Токен привязан к IP адресу, на который он выдан

Для серии запросов можно использовать “execute”.

[Подробнее в документации.](#)

Рассматриваем демо-приложение



Hello, Go

❏ <https://github.com/Quasilyte/hello-go>

Очень компактный сборник материалов и FAQ по Go для начинающих.



Pull Requests и пожелания приветствуются.

Устанавливаем hello-go

Выполните команды в терминале:

```
# Ваш access_token  
export TOKEN='...'
```

```
# Скачиваем hello-go  
go get -v github.com/quasilyte/hello-go/...
```

Устанавливаем hello-go

Выполните команды в терминале:

```
# Очень простое приложение, работающее со storage  
vk-storage -help
```

```
# Программа чуть посложнее, работает с friends  
vk-friends -help
```

Если vk-friends (или vk-storage) “не найден”

Выполните команду “go env GOPATH”.

Вам должна быть выведена директория.

Внутри этой директории есть папка “bin”.

Эту папку нужно добавить в свой системный PATH.

Либо использовать vk-friends прямо оттуда.

Запускаем команды vk-storage

Выполните команды в терминале:

```
$ vk-storage -token $TOKEN set foo spotty  
1
```

```
$ vk-storage -token $TOKEN get foo  
spotty
```

Запускаем команды vk-friends

Выполните команды в терминале:

```
$ vk-friends -token $TOKEN online  
friends online (24):  
... (список друзей онлайн)
```

```
$ vk-friends -token $TOKEN list  
friends (140):  
... (полный список друзей)
```

Весь исходный код в `cmd/vk-*`

А мы разберём лишь некоторые важные части



code overview

Используемые библиотеки

`flag` - работа с `command-line` флагами (аргументами).
`fmt` - форматированный I/O.
`log` - минималистичный пакет логгирования.
`net/url` - библиотека для работы с URL.
`net/http` - HTTP клиент и сервер.
`strings` - функции и типы для работы со строками.
`encoding/json` - сериализация/десериализация JSON.
`io/ioutil` - полезняшки для чтения/записи.
`time` - работа со временем, таймерами.

code overview

interface{} - аналог any/Object/void*

```
// Мы используем один тип для всех  
// ответов API, потому что альтернативой является  
// заведение отдельных типов под каждый вид запроса,  
// что гораздо проще делать кодогенерацией,  
// но не очень приятно делать вручную.
```

```
type vkResponse map[string]interface{}
```

code overview

В vk-friends показан удобный паттерн “[steps](#)”

```
// Не используйте log.Fatal.  
// Если нужен грязный 1-liner, panic лучше.  
// А ещё лучше просто log.Print + return.  
  
~if err := args.parse(); err != nil {  
-   log.Fatalf("parse args error: %v", err)  
+   log.Panicf("parse args error: %v", err)  
~}
```

code overview

Старайтесь делать “контекст” явным

```
// Предпочитайте flag.*Var функции.  
// Для всех аргументов можно завести отдельный тип.  
// Избегайте флагов через глобальные переменные.
```

```
-verbose := flag.Bool("verbose", false, ...)  
+var args arguments  
+flag.BoolVar(&args.verbose, "verbose", false, ...)
```

Делаем своё приложение



Какая-то тактика

1. Придумайте идею
2. Сделайте минимальный, простейший прототип
3. Если остаётся время, мы его доработаем!

Идея не обязана быть уникальной среди всех участников. Главное - чтобы что-то работало.

Идея [0]

Программа, которая добавляет вас в группу
golang_nizhny (или gdgnizhny)



Идея [1]

Ставит лайки всему, чему может, где есть хеш-тег
#golang (и #golang_nizhny)



Идея [2]

Удаляет сообщения со стены, в которых есть
указанных хеш-тег



Идея [3]

Отправляет самому себе сообщение с задержкой
(почти напоминалка)



Идея [4]

<https://habr.com/ru/post/449656/>

(Как развлечь себя с помощью VK API)



Ну, вы поняли, лучше вам самим придумать :)



Приступаєм!

Піднімайте руку, якщо у вас виникли труднощі



Подарочки!



- VK Tech Стикеры всем участникам
- Стикер паки за работающее приложение
- Супер подарок за лучшее приложение

Ссылки/контакты

<https://vk.com/gdgnizhny> - GDG в Нижнем

https://vk.com/golang_nizhny - Golang в Нижнем

<https://quasilyte.dev/gopherkon/> - Гоферы!

<https://vk.com/tech> - о технологиях ВКонтакте

<https://github.com/vkcom> - VK на GitHub