

← Учебный лагерь APG1. День06

Отзы

Submit the project

Finish project

Лагерь выживания

[Показать все](#)

Учебный лагерь APG1. День00	✓ 100 %
Учебный лагерь APG1. День01	✓ 100 %
Учебный лагерь APG1. День02	✓ 100 %
Учебный лагерь APG1. День03	✗ 0 %
Учебный лагерь APG1. День04	✓ 100 %
Учебный лагерь APG1. Команда00	✗ 0 %
Учебный лагерь APG1. День05	
Учебный лагерь APG1. День06	
Учебный лагерь APG1. День07	
Учебный лагерь APG1. День08	
Учебный лагерь APG1. День09	

Частный проект Git

ssh://git@repos-ssh.21-school.ru:2289/students/Go\_Day06.ID\_376227/mlarra\_student.21\_scho...

Копировать ссылку

Открыть

## Задача

# День 06. Учебный лагерь

## Крепость одиночества

### Содержание

1. Глава I
  - 1.1. Основные правила
2. Глава II
  - 2.1. Правила дня
3. Глава III
  - 3.1. вступление
4. Глава IV
  - 4.1. Упражнение 00: Удивительный логотип
5. Глава V
  - 5.1. Упражнение 01. Место для ваших мыслей
6. Глава VI
  - 6.1. Упражнение 02: Ненавистники будут ненавидеть

### Глава I

#### Основные правила

- Ваши программы не должны завершаться неожиданно (выдавая ошибку при правильном вводе). Если это произойдет, ваш проект будет считаться неработоспособным и получит 0 баллов при оценке.
- Мы рекомендуем вам создать тестовые программы для вашего проекта, даже если эту работу не нужно будет отправлять и она не будет оцениваться. Это даст вам возможность легко проверить свою работу и работу ваших коллег. Вы найдете эти тесты особенно полезными во время вашей защиты. Действительно, во время защиты вы можете использовать свои тесты и/или тесты коллеги, которого вы оцениваете.
- Отправьте свою работу в назначенный репозиторий git. Оцениваться будет только работа в репозитории git.
- Если ваш код использует внешние зависимости, он должен использовать модули Go для управления ими.

### Глава II

#### Правила дня

- Вы должны только сдать \*.go файлы и (в случае внешних зависимостей) go.mod + go.sum
- Ваш код для этой задачи должен быть собран с помощью всего лишь go build
- Дополнительные шаги, например, для создания таблиц в базе данных, должны быть включены в admin\_credentials.txt .

## Глава III

---

### вступление

---

Итак - это оно! У тебя просто суперспособности! И, как новый супергерой, вам обязательно нужно подумать о своем уличном авторитете и всеобщем признании, не так ли?

Огненные шары - проверить! Почти до неприличия обтягивающее трико - проверено!  
Секретная база - о да, детка!

Что-нибудь еще вы забыли? Любые другие махинации, чтобы выполнить? О, ПОДОЖДИТЕ~

## Глава IV

---

### Упражнение 00: Удивительный логотип

Единственное требование здесь — создать потрясающий логотип! Это должен быть PNG-файл размером 300x300 пикселей с именем «amazing\_logo.png». Сделайте это так круто, как вы можете!

Изображение должно появиться в том же каталоге, что и запущенный двоичный исполняемый файл после компиляции.

NOTE: you shouldn't cheat and download anything from the internet in your code. Just don't rely on any external sources and unleash your fantasy in programming! Also, just plain single color or transparent logo doesn't count!

## Chapter V

---

### Exercise 01: Place for Your Thoughts

Okay, so what else do you need now? A website, of course! It should be a blog where everybody will be able to read your ideas on the world improvement. Here is a list of features it should have:

- Database (you should use PostgreSQL)
- Admin panel (on '/admin' endpoint) where only you can login with just a form for posting new articles (let's forget about editing old ones for now, superheroes don't ever look back)
- Basic markdown support (so it can at least show "###" headers and links in generated HTML)

- Pagination (show no more than 3 thoughts on one page for people to not get too much of your awesomeness)
- Application UI should use port 8888

All additional files (images, css, js if you decide to use any of those) should be submitted as a *zip* file to be unpacked in the same directory as binary itself, resulting into something like this: .

```
├─ css
│   └─ main.css
├─ images
│   └─ my_cat.png
├─ js
│   └─ scripts.js
└─ myblog-binary
```

Admin credentials for posting access (login and password) and database credentials (database name and user) should be submitted separately as well in a file called *admin\_credentials.txt*. If there are additional commands to be run to create tables in a database, put them into the same file.

Main page should include your logo from EX00, links to articles and (optionally) some short preview of their content, as well as pagination (if there are more than 3 articles in a database).

When clicking a link to article, user should get to a page with a rendered markdown text and a single "Back" link which brings him/her back to main page.

## Chapter VI

### Exercise 02: Haters Gonna Hate

Now, when you already have a cool website, let's update it a little and protect ourselves from the villains trying to bring it down! All you need to do is implement rate limiting, so if there are more than a hundred clients per second trying to access it, they should get a "429 Too Many Requests" response. Of course when you're getting more famous we'll raise that limit! It's just for testing for now.