

Описание репозитория telegramBotSlivYandexLuceum Данный репозиторий (<https://github.com/glortest/telegramBotSlivYandexLuceum>) содержит код для телеграм-бота, предназначенного для решения задач и хранения данных о пользователях, купивших премиум аккаунт. В репозитории присутствуют файлы с расширениями .json и .txt, которые содержат соответствующие данные.

Файлы с расширением .json Файлы с расширением .json содержат "решения задач". Они предположительно хранят информацию о задачах, их описании, решениях и других деталях. Формат данных в файлах .json представлен в виде структуры, согласно синтаксису JSON (JavaScript Object Notation).

Пример содержимого файла с расширением .json:

```
json Copy code
```

Файлы с расширением .txt Файлы с расширением .txt содержат данные о пользователях, купивших премиум аккаунт. Они, вероятно, содержат информацию о пользователях, такую как идентификаторы, даты покупки, длительность премиум статуса и другую информацию. Формат данных в файлах .txt предполагается текстовым и может быть структурирован в определенном формате.

Пример содержимого файла с расширением .txt:

Телеграмм-бот решений программы "Яндекс.Лицей". Первый год обучения

Основная идея проекта и цель:

Создать Телеграмм бота, позволяющего получать решения задач программы Яндекс.Лицей первого года обучения.

Техническое задание:

- Создать базу решений задач
- Создать Telegramм-бот
- Проработать механизм выдачи решения задач в боте
- Реализовать возможность монетизации проекта
- Доделать проект до готового вида (Загрузить на github, написать документацию и тд.)

Устройство проекта

Используемые библиотеки

```
sys
json
telegram.ext
logging
```

Язык:

```
Python 3.9
```

Можно использовать 3.10, но на него не написана библиотека телеграмм

Хранение данных

При написании проекта, перед нами сразу же встал вопрос: *А как хранить решения?* Неужели придётся всё записать в словарь? Немного поразмышляв, мы пришли к идее использовать формат файла ".json" для хранения решений - работа с ним очень схожа с взаимодействием со словарём в языке python. Таким образом, в файле у нас образовалось два типа хранения данных:

.json

Здесь хранятся решения задач. Таких файла два: в db_free.json - решения, доступные без премиум подписки, в db_cash.json доступные только с ней.

```
{"task_name": "Task",}
```

.txt

В файле CashUsers.txt хранится информация о пользователях, купивших премиум, в Request.txt - запросивших премиум.

```
"1295434486Dunk00сипенко ЕвдокияNone"
```

Код

Методы:

```
ASYNC DEF READINGNOTIFICATIONS(UPDATE, CONTEXT)
```

Проверка на приём новых уведомлений.

```
DEF CHECKINGFORPREMIUM(MESSAGE)
```

Проверка на владение пользователем "Премиум" аккаунта.

```
DEF CHECKINGAVAILABILITY(TEXT, PREMIUM):
```

Проверка уведомлений на команды и необходимый нам текст.

```
DEF MAIN()
```

Ну main как-бы, запуск программы, база, основа.

Команды бота

/start - старт /help - помощь /vip - инструкция получения vip

Чтобы получить решение задачи, необходимо отправить её название в чат бота.

Также есть команды для разработчиков