

Telegram Бот

@eassssy_learn_bot

Идея

Каждый день, пользователю будут опраляться интересные факты из разных отраслей, а после прочтения представленной информации он сможет пройти небольшой тест для повторения прочитанного. Человек может сам выбрать язык, на котором будет представлена информация.

В дальнейшем, планируется автоматизация процесса поиска информации для пользователей и составление тестов, преимущественно этот процесс будет проходить с помощью нейросетей.

Библиотеки

- telebot - основная библиотека используемая в программе, служит для обработки/отправки сообщений пользователя, регистрации токена бота
- random - выбор случайной темы для текста
- googletrans - библиотека, реализующая API Google Translate
- asyncio - я использовала для работы с translator():

Реализация

@bot.message_handler(commands=['...']) - обработчик команд, которые отправляются по нажатию на кнопки

Непосредственно создание кнопок (в этом примере 2 кнопки, можно больше):

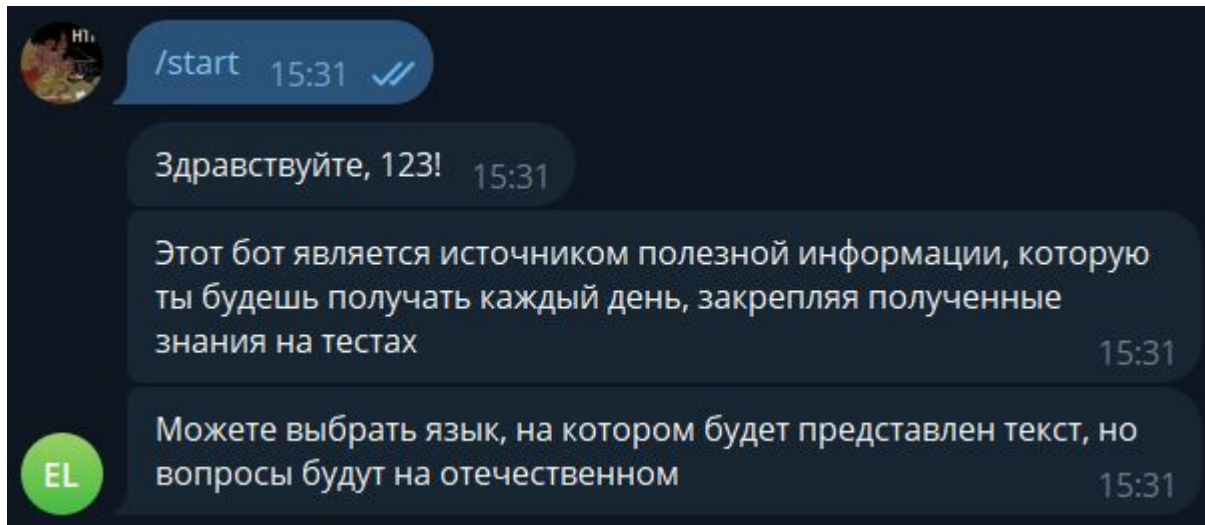
markup = types.ReplyKeyboardMarkup(resize_keyboard=True) - объявление

btn1 = types.KeyboardButton('...')

btn2 = types.KeyboardButton('....')

markup.add(btn1, btn2) - добавление

bot.send_message(message.chat.id, f'...?', reply_markup=markup) -
отправляет сообщение, на которое пользователь будет отвечать кнопками



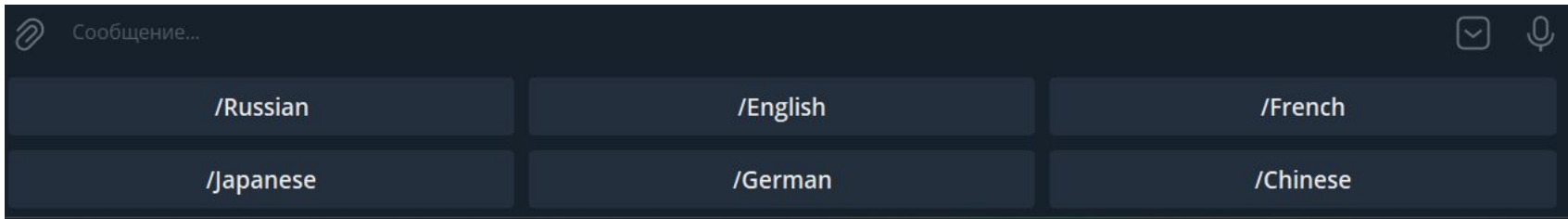
`bot.register_next_step_handler(message, qx4)` - очень полезная функция `telebot`, которая запускает вашу собственную функцию/метод только после того, как пользователь отправил любое сообщение

from googletrans import Translator

Немного про эту библиотеку: Она делает AJAX-вызовы к API, чтобы определять языки и переводить текст.

AJAX-вызовы (AJAX — Asynchronous JavaScript and XML) - это технология для веб-разработки, которая позволяет обновлять содержимое страницы без полной перезагрузки

```
async def tranz():  
    global hex, prev  
    async with Translator() as translator:  
        result = await translator.translate(prev,  
            src="ru", dest=f'{hex}')  
        bot.send_message(message.chat.id,  
            result.text)  
    asyncio.run(tranz())
```



Итог

- В дальнейшем можно сделать опросник, который будет предлагаться в начале использования бота, который определяет наиболее интересные для пользователя темы и присылать интересную информацию уже по ним
- Добавить больше визуальной части: высылать gif или стикеры с картинками после выполнения теста в зависимости от успешности выполнения
- Монетизация - найти рекламодателей и высылать в чат персонализированную рекламу

В ходе разработки этого проекта я закрепила свои знания в создании telegram-бота и узнала о больших возможностях в работе с библиотекой telebot