import vk\_api

from vk\_api.utils import get\_random\_id

import go

from bs4 import BeautifulSoup

import requests

from random import \*

import lxml

# импортируем токен бота

vk = vk\_api.VkApi(token="fe5a7313b94f87d7856fd4f628108d0ec175155033c0f29d96021784b668f92f948457e3833a5798a5c00")

sl = ''

first = False

first\_for\_goroda = False

def question():

global id

varies = ['ДА', 'НЕТ']

vk.method("messages.send",

{'peer\_id': id, 'message': varies[randint(0, 1)],

'random\_id': get\_random\_id(), 'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

def goroskop():

global id

r = requests.get('https://1001goroskop.ru/')

soup = BeautifulSoup(r.text, 'lxml')

vk.method("messages.send",

{'peer\_id': id, 'message': 'Гороскоп был взят с сайта 1001goroskop.ru',

'random\_id': get\_random\_id()})

vk.method("messages.send",

{'peer\_id': id, 'message': soup.select\_one('[itemprop="description"]').text,

'random\_id': get\_random\_id(), 'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

def viselica():

global text, id, word, guess, life, listused, win

if life:

if text.lower() in listused:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Вы уже использовали эту букву',

'random\_id': get\_random\_id()})

elif text.lower() in word:

for w in range(len(word)):

if text.lower() == word[w]:

guess[w] = text.lower()

if word == ''.join(guess):

win = True

listused.append(text.lower())

elif text.lower() not in word:

life -= 1

listused.append(text.lower())

if life:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Слово: ' + ' '.join(guess),

'random\_id': get\_random\_id()})

if not win:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Использованнные буквы: [' + ', '.join(listused) + ']',

'random\_id': get\_random\_id()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'У вас осталось ' + '&#10084;' \* life + ' жизней',

'random\_id': get\_random\_id(), 'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'Введите пожалуйста букву',

'random\_id': get\_random\_id()})

def gorodagame():

global messages, text, first\_for\_goroda, sl

f = j = False

game\_over = 0

wrong = False

g = text.lower().title()

if first\_for\_goroda:

if sl[-1] == 'ь' or sl[-1] == 'ы' or sl[-1] == 'ъ':

if sl[-2] != g[0].lower():

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неправильный ввод', 'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

wrong = True

else:

if sl[-1] != g[0].lower():

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неправильный ввод', 'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

wrong = True

if not wrong:

if g in go.gor:

go.gor.remove(g)

while f is False and g.lower() != 'cтоп':

if text.lower().title() in go.gor:

go.gor.remove(text.lower().title())

for i in go.gor:

if g[-1] == 'ь' or g[-1] == 'ы' or g[-1] == 'ъ':

if g[-2] == i[0].lower() and f is False:

vk.method("messages.send",

{'peer\_id': id, 'message': i, 'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

f = j = True

sl = i

go.gor.remove(i)

else:

if g[-1] == i[0].lower() and f is False:

vk.method("messages.send",

{'peer\_id': id, 'message': i, 'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

f = j = True

sl = i

go.gor.remove(i)

if j is False and game\_over == 0:

vk.method("messages.send", {'peer\_id': id, 'message': 'Я проиграл', 'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

game\_over += 1

break

first\_for\_goroda = True

def weather():

r = requests.get('https://yandex.ru/pogoda/saint-petersburg') # отправляем запрос на сайт

soup = BeautifulSoup(r.content, 'lxml') # выделяем контент со страницы soup.select(".temp\_\_value") - выборка CSS

vk.method("messages.send",

{'peer\_id': id, 'message': 'Все данные о погоде были взяты с ресурса'

'https://yandex.ru/pogoda/saint-petersburg',

'random\_id': get\_random\_id()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'Сейчас за окном ' + soup.select(".temp\_\_value")[1].text + '°' +

' и ' +

soup.select(".forecast-briefly\_\_condition")[1].text.lower(),

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

if 'дождь' in soup.select(".forecast-briefly\_\_condition")[1].text.lower():

vk.method("messages.send",

{'peer\_id': id, 'message': 'Зонтик не будет лишним)',

'random\_id': get\_random\_id()})

if 'пасмурно' in soup.select(".forecast-briefly\_\_condition")[1].text.lower():

vk.method("messages.send",

{'peer\_id': id, 'message': 'Возможен дождь...',

'random\_id': get\_random\_id()})

if 'снег' in soup.select(".forecast-briefly\_\_condition")[1].text.lower():

vk.method("messages.send",

{'peer\_id': id, 'message': 'Одежды много не бывает, одентесь теплее)',

'random\_id': get\_random\_id()})

def health():

global id

total = ''

vk.method("messages.send",

{'peer\_id': id, 'message': 'Пожалуйста введите ваш вес и рост через пробел(только числа)',

'random\_id': get\_random\_id(), 'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

while True:

# сортируем непрочитанные сообщения

messages = vk.method("messages.getConversations", {'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

id = messages['items'][0]['last\_message']['from\_id'] # выделяем из сообщения id пользователя

text = messages['items'][0]['last\_message']['text']

if ' ' in text and all(tt.isdigit() for tt in text.split()) or text.lower() == 'стоп':

break

else:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неправильный ввод',

'random\_id': get\_random\_id(), 'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

if text.lower() != 'стоп':

weight, height = map(float, text.split())

height /= 100

bmi = round(weight / (height \*\* 2), 2)

if bmi < 16:

total = 'Выраженный дефицит массы тела'

if 16 <= bmi < 18.5:

total = 'Недостаточная масса тела'

if 18.5 <= bmi < 25:

total = 'Нормальная масса тела'

if 25 <= bmi < 30:

total = 'Избыточная масса тела (предожирение)'

if 30 <= bmi < 35:

total = 'Ожирение 1-ой степени'

if 35 <= bmi < 40:

total = 'Ожирение 2-ой степени'

if bmi >= 40:

total = 'Ожирение 3-ей степени'

vk.method("messages.send",

{'peer\_id': id, 'message': 'У вас ' + str(bmi) + ' BMI - ' + str(total),

'random\_id': get\_random\_id(), 'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

else:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Остановлено',

'random\_id': get\_random\_id(), 'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

while True:

# сортируем непрочитанные сообщения

messages = vk.method("messages.getConversations", {'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

if messages['items'][0]['last\_message']['from\_id'] != id:

first = False

id = messages['items'][0]['last\_message']['from\_id'] # выделяем из сообщения id пользователя

text = messages['items'][0]['last\_message']['text'] # содержание сообщения(именно текст)

if text.lower() == 'игра':

vk.method("messages.send", {'peer\_id': id, 'message': 'Выберите пожалуйста игру',

'random\_id': get\_random\_id(),

'keyboard': open('game.json', "r", encoding="UTF-8").read()})

while True:

messages = vk.method("messages.getConversations", {'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

text = messages['items'][0]['last\_message']['text']

if text.lower() == 'стоп':

vk.method("messages.send",

{'peer\_id': id, 'message': 'Остановлено',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

break

if text.lower() == 'виселица':

win = 0

vk.method("messages.send", {'peer\_id': id, 'message': 'Добро пожаловать в игру Виселица. '

'Выберите тему игры',

'random\_id': get\_random\_id(),

'keyboard': open('theme.json', "r", encoding="UTF-8").read()})

life = 2

listused = []

theme = False

dic\_vis = {'спорт': ['борьба', 'олимпиада', 'клюшка', 'шайба', 'гиря', 'коньки', 'штанга',

'ворота', 'конь', 'козел', 'трамплин', 'болид', 'татами', 'ковер',

'трико', 'борцовки', 'партер', 'трек', 'ринг', 'гонг', 'аут', 'афсайт',

'штрафной', 'пенальти', 'булит', 'проброс', 'фол', 'гол', 'булава',

'тренер', 'голкипер', 'фанат', 'болельщик', 'бровка', 'хоккей',

'футбол', 'гандбол', 'бокс', 'фехтование', 'регби', 'биатлон',

'лыжи', 'гетры', 'баскетбол', 'волейбол', 'теннис', 'ракетка',

'валан', 'тренажер', 'велосипед', 'канат', 'гольф', 'сетка',

'арбитр', 'рефери', 'обруч', 'мат', 'батут', 'скакалка', 'тир',

'стадион', 'каток', 'молот'],

'учеба': ['математика', 'учитель', 'класс', 'парта', 'портфель', 'ученик', 'звонок',

'одноклассник', 'ручка', 'доска', 'урок', 'тетрадь', 'учебник']}

dict\_ru = ['а', 'б', 'в', 'г', 'д', 'е', 'ё', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п',

'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я']

while True:

if win is True:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Вы выиграли!',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

break

else:

messages = vk.method("messages.getConversations",

{'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

text = messages['items'][0]['last\_message']['text']

if text.lower() != 'стоп':

if not theme and text.lower() in dic\_vis:

word = dic\_vis[text.lower()][randint(0, len(dic\_vis[text.lower()]) - 1)]

guess = ['-'] \* len(word)

vk.method("messages.send",

{'peer\_id': id, 'message': ' '.join(guess),

'random\_id': get\_random\_id()})

vk.method("messages.send",

{'peer\_id': id,

'message': 'Длина слова ' + str(len(word)) + ' букв',

'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'Введите пожалуйста букву',

'random\_id': get\_random\_id()})

theme = True

elif not theme and text.lower() not in dic\_vis:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неизвестная команда',

'random\_id': get\_random\_id(),

'keyboard': open('theme.json', "r", encoding="UTF-8").read()})

if theme and text.lower() not in dic\_vis:

if len(text) > 1 or not text.isalpha() or text.lower() not in dict\_ru:

vk.method("messages.send",

{'peer\_id': id,

'message': 'Неправильный ввод, введите одну букву',

'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

else:

viselica()

if life == 0:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Вы проиграли',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r",

encoding="UTF-8").read()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'Слово: ' + word,

'random\_id': get\_random\_id()})

break

else:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Конец игры',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

vk.method("messages.send",

{'peer\_id': id, 'message': 'Слово: ' + word,

'random\_id': get\_random\_id()})

break

break

elif text.lower() == 'города':

vk.method("messages.send",

{'peer\_id': id, 'message': 'Добро пожаловать в игру "Города России". Вы начинаете',

'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

while True:

messages = vk.method("messages.getConversations",

{'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

text = messages['items'][0]['last\_message']['text']

if text.lower() != 'стоп' and len(text) > 1 and (text.title() in go.gor or text in go.gor):

gorodagame()

elif text.lower() == 'стоп':

vk.method("messages.send",

{'peer\_id': id, 'message': 'Конец игры',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

break

elif len(text) <= 1 or text.title() not in go.gor:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неправильный ввод',

'random\_id': get\_random\_id(),

'keyboard': open('stop.json', "r", encoding="UTF-8").read()})

break

else:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неизвестная команда',

'random\_id': get\_random\_id(),

'keyboard': open('game.json', "r", encoding="UTF-8").read()})

elif text.lower() == 'случайный ответ':

vk.method("messages.send",

{'peer\_id': id, 'message': 'Пожалуйста введите вопрос, на который можно ответить "да" или "нет". '

'В конце вопроса должен быть "?"',

'random\_id': get\_random\_id()})

while True:

messages = vk.method("messages.getConversations", {'offset': 0, 'count': 20, 'filter': 'unread'})

if messages['count'] >= 1:

text = messages['items'][0]['last\_message']['text']

if text[-1] != '?' and len(text) > 1:

vk.method("messages.send",

{'peer\_id': id,

'message': 'Неправильный ввод',

'random\_id': get\_random\_id()})

else:

question()

break

elif text.lower() == 'погода':

weather()

elif text.lower() == 'гороскоп':

goroskop()

elif text.lower() == 'здоровье':

health()

elif text.lower() != '' and first is False:

vk.method("messages.send", {'peer\_id': id, 'message': 'Доброго времени суток', 'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})

first = True

else:

vk.method("messages.send",

{'peer\_id': id, 'message': 'Неизвестная команда',

'random\_id': get\_random\_id(),

'keyboard': open('keyboard.json', "r", encoding="UTF-8").read()})