

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення
Дисципліна: Скриптові мови програмування (Python)

Лабораторна робота №9

**Тема: «ВИКОРИСТАННЯ СТОРОННІХ API ДЛЯ ЧАТ-БОТУ В
TELEGRAM ЗАСОБАМИ МОВИ РУТНОН»**

Виконав: ст. гр. КН-24
Куріщенко П. В.
Перевірив: асистент
Ткаченко О.С.

Кропивницький 2025

Мета роботи - навчитися використовувати готові реалізації оболонок API для чат-ботів на Python.

ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ

Завдання 1:

Реалізуйте приклад, вказаний у даній лабораторній роботі.



Опис Telegram-бота "Погода"

Цей Telegram-бот надає користувачу інформацію про поточну погоду в заданому місті, використовуючи API сервісу **OpenWeatherMap** через бібліотеку **pyowm**.



Можливості бота:

- Команда **/start** — виводить базове привітальне повідомлення.
- Команда **/city** — запитує в користувача назву міста та повертає:
 - Температуру повітря в градусах Цельсія.
 - Опис погодних умов (ясно, хмарно, дощ і т.д.).
 - Рекомендацію щодо одягу на основі температури.
 - Анімацію (GIF) з реакцією залежно від температури:
 - Менше 0°C — "дуже холодно".
 - Від 0°C до 15°C — "прохолодно".
 - Більше 15°C — "тепло".



Використані бібліотеки:

- telebot — для створення Telegram-бота.
- pyowm — для взаємодії з OpenWeatherMap API.

- tenor GIFs — для додавання емоційних реакцій на погоду.

Принцип роботи:

1. Користувач вводить команду /city.
2. Бот запитує назву міста.
3. Бот отримує погодні дані з API.
4. Виводить інформацію + GIF-анімацію + пораду щодо одягу.

Лістинг бота:

```
import telebot
from safe import exm_token
from pyowm import OWM
from pyowm.utils.config import get_default_config

bot = telebot.TeleBot(exm_token)
config_dict = get_default_config()
config_dict['language'] = 'ua'
owm = OWM( '0f8480e94c08f6fa663ead777cd2ee53', config_dict )

@bot.message_handler(commands=['start']) #реакція чат бота на стартову команду
def start_message(message):
    bot.send_message(message.chat.id, 'Ваше повідомлення /start')

@bot.message_handler(commands=['city']) #команда для отримання початкових
даних
def cmd_city(message):
    send = bot.send_message(message.chat.id, 'Введи місто')
    bot.register_next_step_handler(send, city)

def city(message): #основна робота з декількома варіантами в залежності від
результату
    bot.send_message(message.chat.id, 'Дізнаюсь погодні умови в місті
{city}'.format(city=message.text))
    data = message.text
    mgr = owm.weather_manager()
    observation = mgr.weather_at_place(data)
    observation = mgr.weather_at_place(message.text)
    w = observation.weather
    t = w.temperature('celsius')['temp']

    answer = f"В місті {message.text} зараз {w.detailed_status} \n"
    bot.send_message(message.chat.id, answer)
```

```

answer += f"Приблизна температура {round(t)} °C\n\n"

if t < 0:
    answer += 'Зараз температура нижче нуля, одягайся тепліше!'
    bot.send_animation(message.chat.id,
'https://media1.tenor.com/m/Xv92ltSnpgAAAAAd/gosling-sad-gosling.gif')
elif t < 15:
    answer += 'Зараз прохолодно, варто тепліше одягтися!'
    bot.send_animation(message.chat.id,
'https://media1.tenor.com/m/tgLuPl06kz8AAAAAd/what-gosling.gif')
else:
    answer += 'Зараз досить тепло, можна одягтися легко!'
    bot.send_animation(message.chat.id,
'https://media1.tenor.com/m/F1z461rxQdwAAAAAd/gosling-barbie.gif')

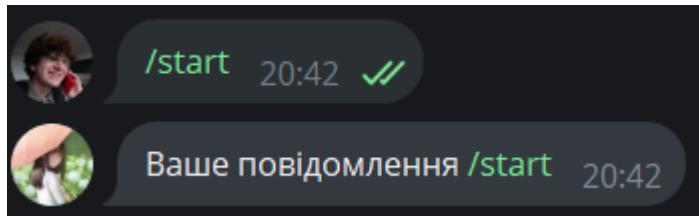
bot.send_message(message.chat.id, answer)

bot.polling(none_stop=True) # необхідно, щоб бот не вимкнувся одразу

```

Результати

роботи:





/city 20:52 ✓



Введи місто 20:52



Олександрія 20:52 ✓

Дізнаюсь погодні умови в місті Олександрія 20:52



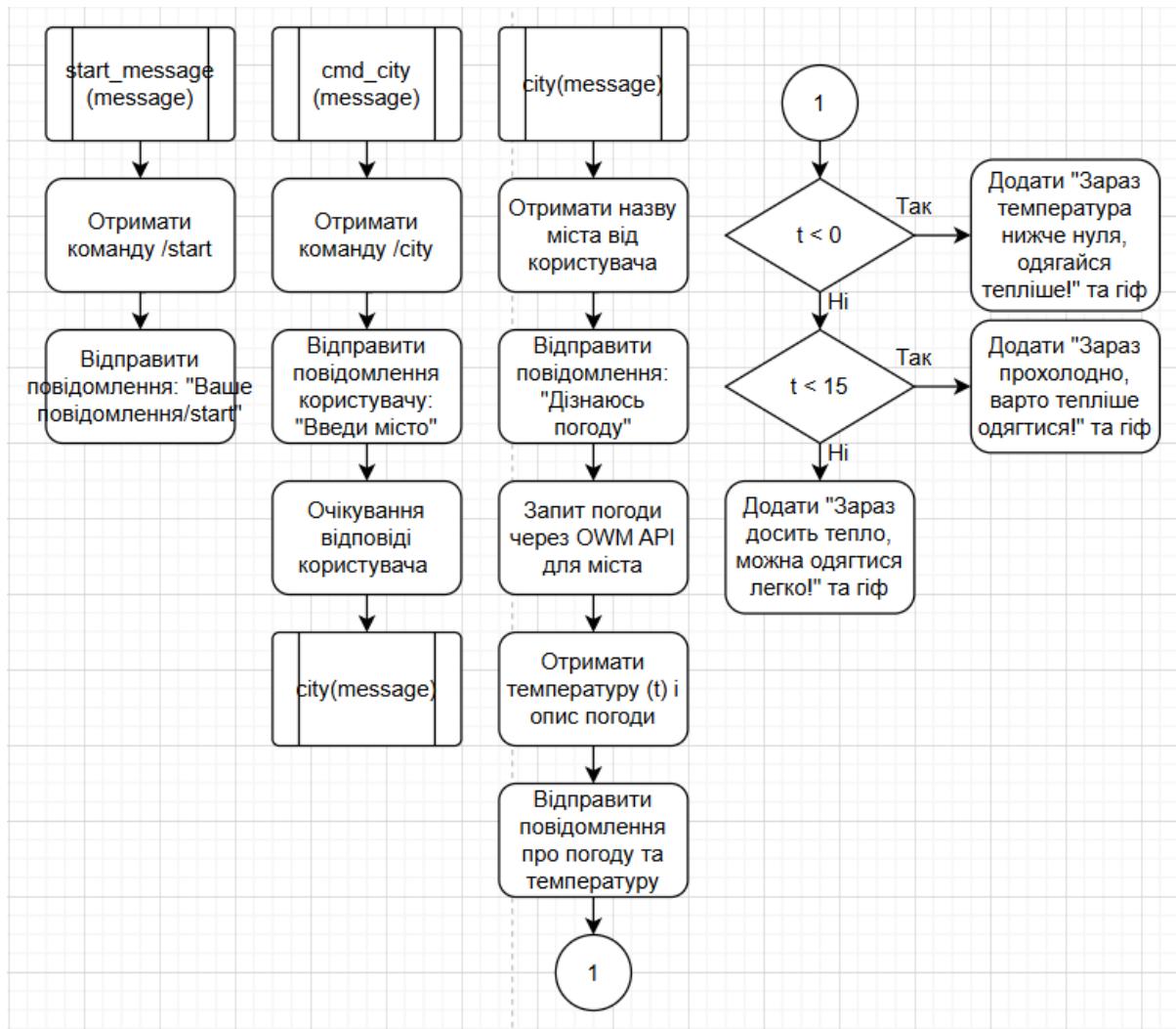
В місті Олександрія зараз рвані хмари
Приблизна температура 9 °C



Зараз прохолодно, варто тепліше одягтися! 20:52

Блок

схеми:



Завдання 2:

Оберіть для себе інший web APIs, ніж у прикладі, з яким хочете працювати та реалізувати його функції в варіанті Вашого бота.



Опис роботи TikTok Downloader бота

Цей Telegram-бот дозволяє користувачу **завантажувати відео** з **TikTok** без водяного знаку. Все, що потрібно — це надіслати йому посилання на відео з TikTok.



Основні можливості:

1. Команда /start

Відповідає вітальним повідомленням: "Привіт! Надішли мені посилання на відео з TikTok, і я його скачую 🎥"

2. Обробка посилань на TikTok

Коли користувач надсилає повідомлення, що містить 'tiktok.com', бот:

- 1) Використовує API <https://tikwm.com/api/> для обробки посилання;
- 2) Якщо відео успішно знайдене:

- Отримує пряме посилання на відео (mp4);
- Надсилає відео у відповідь разом з назвою/описом.

- 3) Якщо виникає помилка:

- Надсилає відповідне повідомлення про помилку.

3. Обробка інших повідомлень

Якщо повідомлення не містить TikTok-посилання, бот відповідає:

"👉 Надішли посилання на TikTok відео, щоб я міг його завантажити."

📌 Технології:

- **Мова:** Python

- **Бібліотеки:**

- telebot (для роботи з Telegram API)
- requests (для запиту до стороннього API)
- **API:** [Tikwm.com](https://tikwm.com) — використовується для отримання відео без водяного знаку.

Лістинг бота:

```
import telebot
import requests
from safe import tt_token

bot = telebot.TeleBot(tt_token)

@bot.message_handler(commands=['start'])
def start_message(message):
```

```

bot.send_message(message.chat.id, 'Привіт! Надішли мені посилання на відео
з TikTok, і я його скачаю 🎥')

@bot.message_handler(func=lambda message: 'tiktok.com' in message.text)
def handle_tiktok_link(message):
    url = message.text.strip()
    bot.send_message(message.chat.id, "⏳ Завантажую відео...")

    try:
        api_url = "https://tikwm.com/api/"
        response = requests.get(api_url, params={"url": url})
        data = response.json()

        if data['code'] != 0:
            bot.send_message(message.chat.id, "❌ Не вдалося завантажити
відео. Спробуй інше посилання.")
            return

        video_url = data['data']['play'] # посилання на mp4
        caption = data['data']['title']

        # Надсилання відео користувачу
        bot.send_video(message.chat.id, video=video_url, caption=caption)

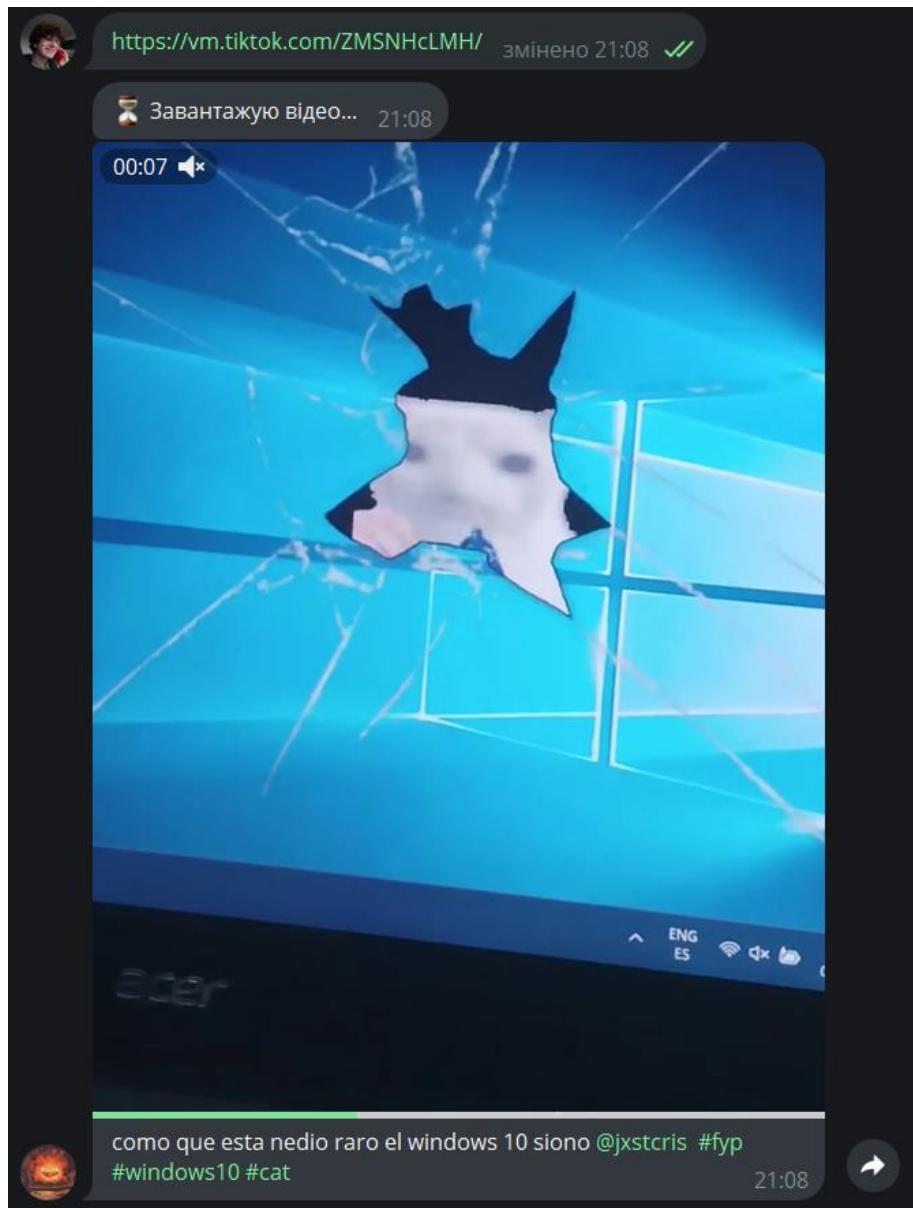
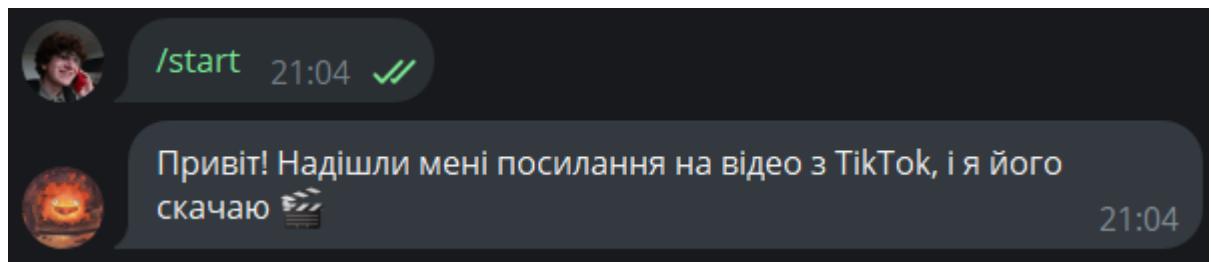
    except Exception as e:
        bot.send_message(message.chat.id, f"⚠️ Сталася помилка: {e}")

@bot.message_handler(func=lambda message: True)
def fallback(message):
    bot.send_message(message.chat.id, "👉 Надішли посилання на TikTok відео,
щоб я міг його завантажити.")

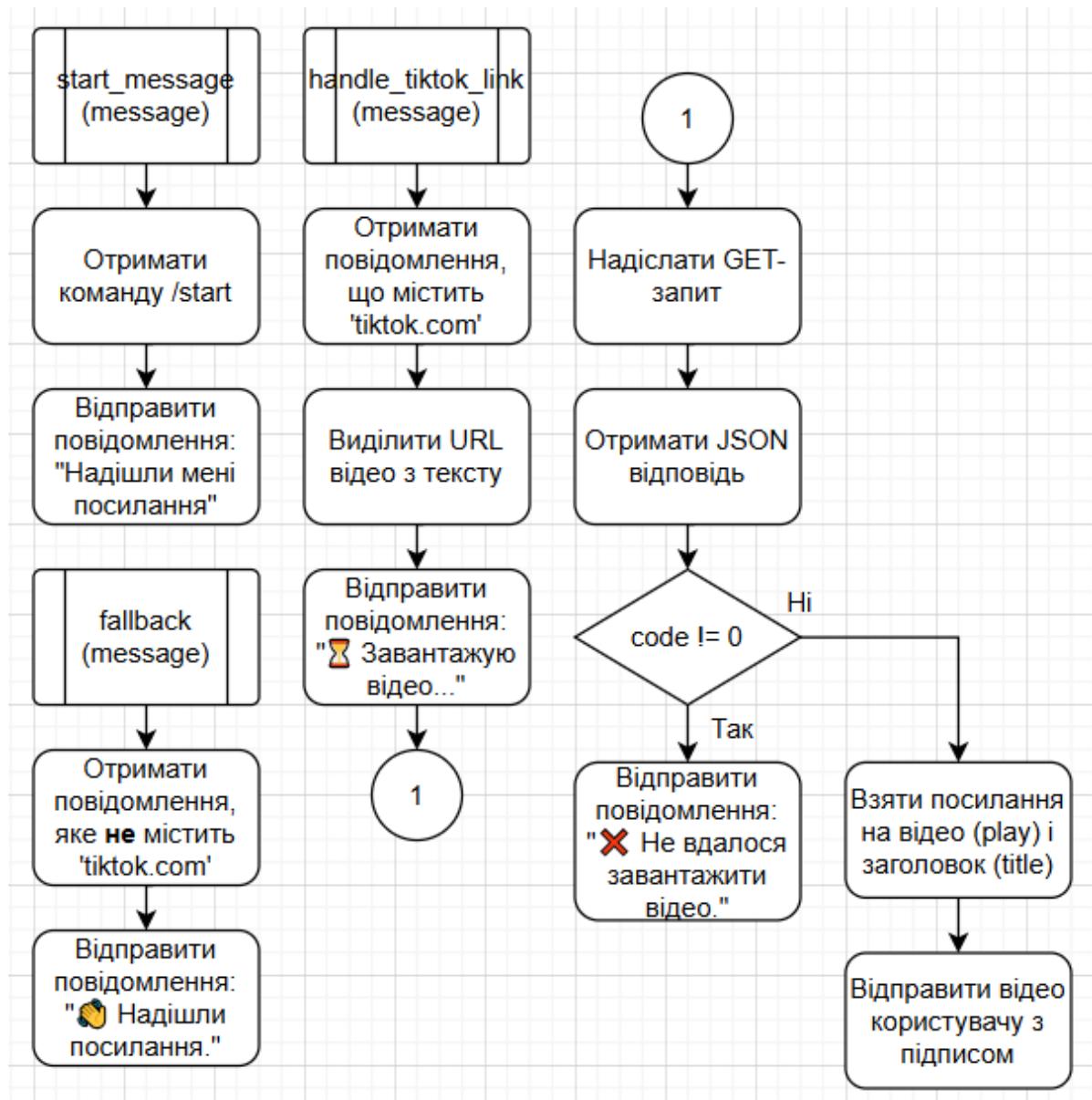
bot.polling(none_stop=True)

```

Результат роботи:



Блок схеми:



КОНТРОЛЬНІ ЗАПИТАННЯ

1. Для чого використовується бібліотека pyowm?

Для отримання актуальної погоди з OpenWeatherMap API.

2. Які нові функції Ви додали чат-боту для телеграм?

Завантаження відео з TikTok за посиланням, обробка неправильних повідомлень, реакція на /start.

3. Для чого використовується символ @ у наведеному прикладі коду?

Це декоратор, який використовується для позначення обробників повідомлень бота (наприклад, команд).

4. Чим буде відрізнятися створення чат-ботів для інших месенджерів?

Відрізняється API, формат повідомлень, авторизація та інструменти розробки.

5. Як можна додати чат-боту елементи штучного інтелекту?

Інтегрувати моделі ШІ через API (наприклад, OpenAI) для генерації тексту, відповідей чи аналізу.