МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики» (МТУСИ)**

Кафедра «Системное программирование»

**ОТЧЕТ**

по лабораторной работе № 7 дисциплина **«Введение в информационные технологии»**

Выполнил: Студент гр.БВТ2207

Кузьмин И.А.

“ ” 2023г.

Проверил:

“ ” 2023г.

Москва, 2023

# Введение

# Цель работы:

# Задание:

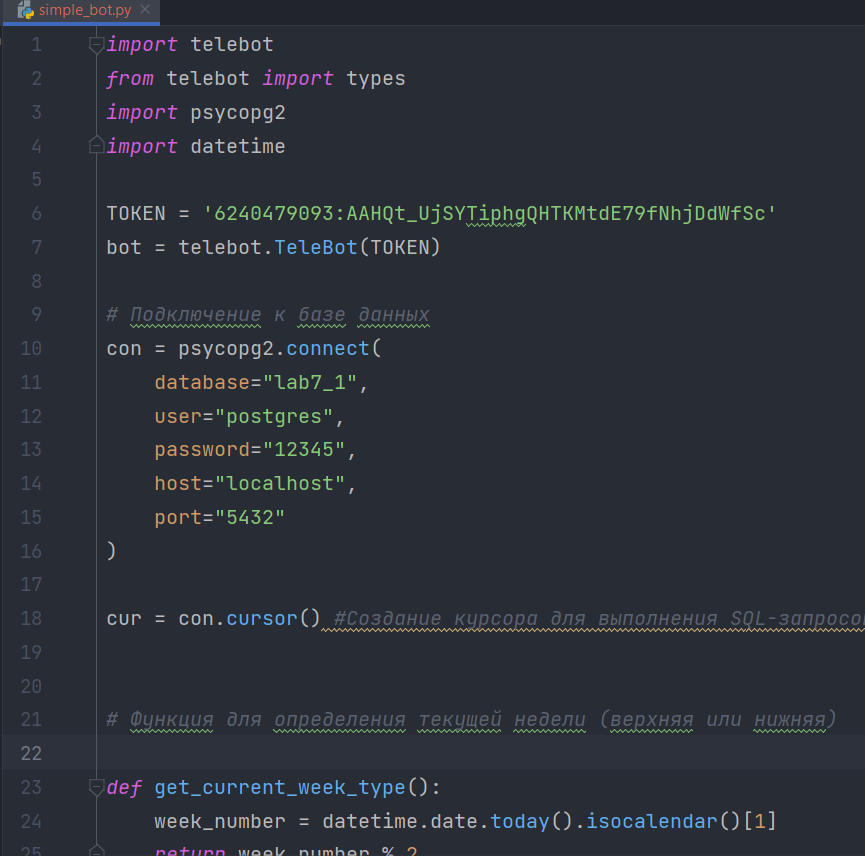
Написать telegram бота с расписанием.

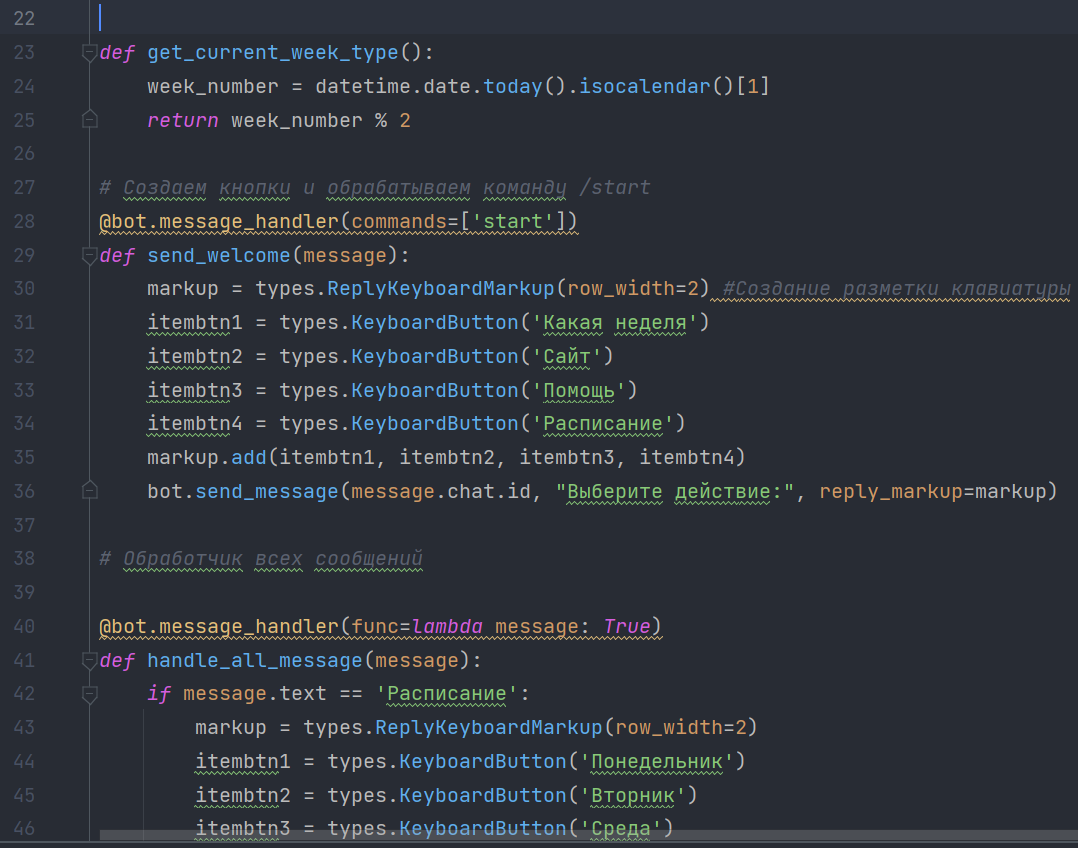
.

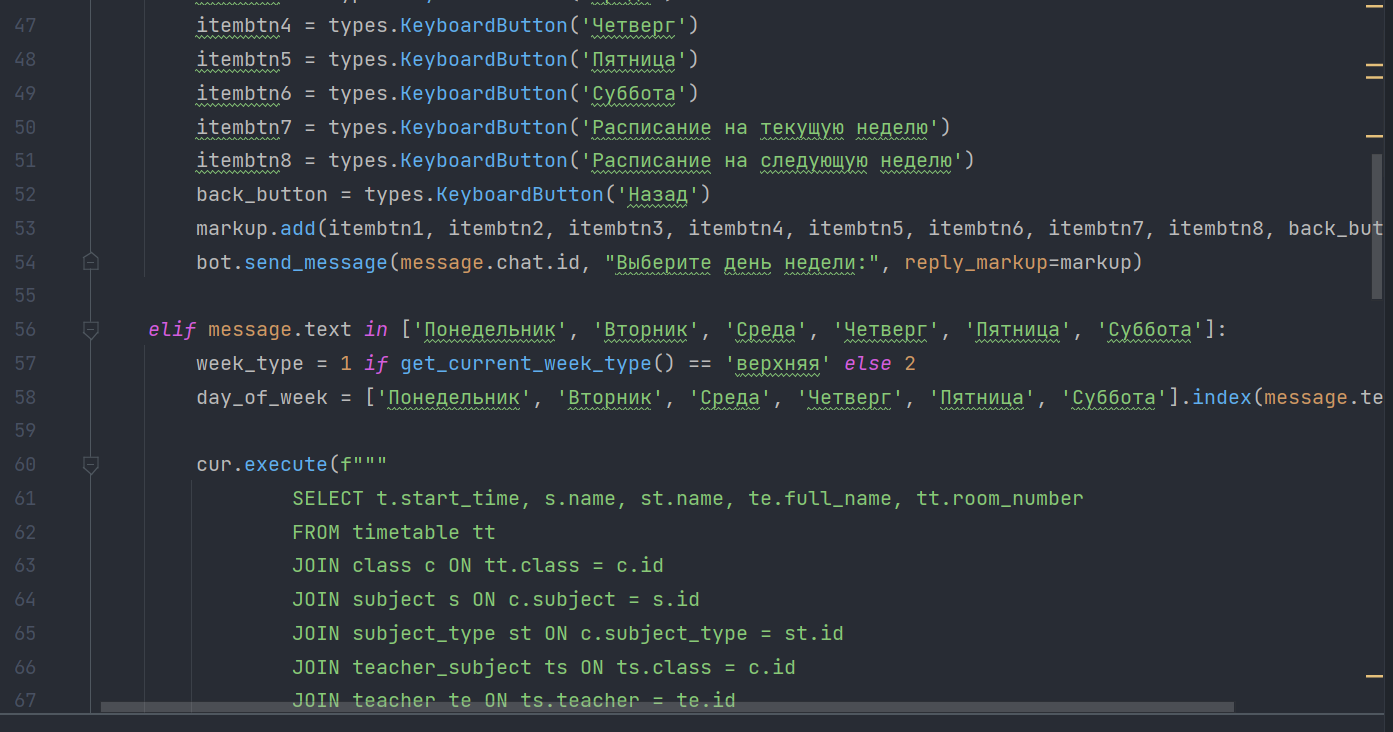
# Реализация

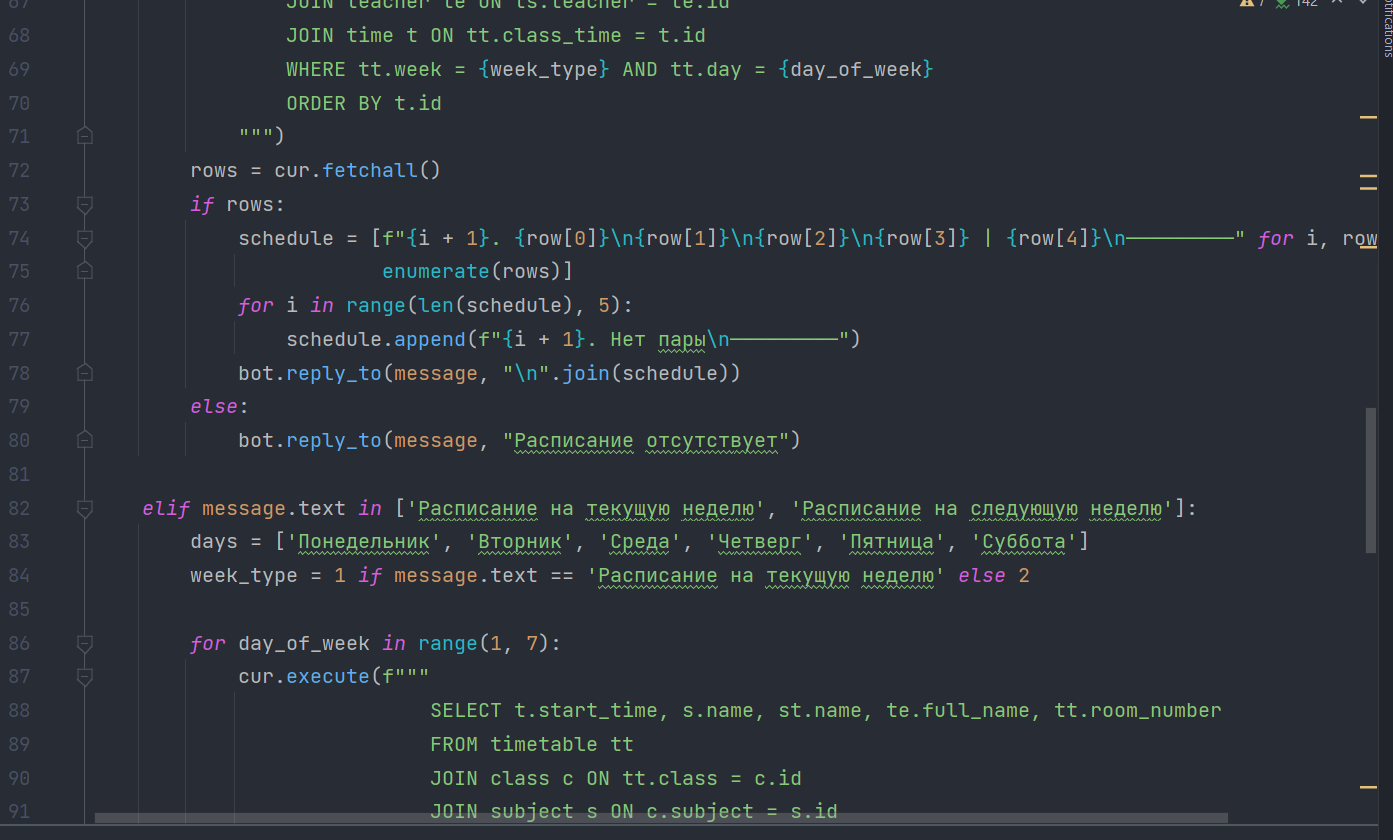
В процессе выполнения лабораторной работы использованы следующие информационные источники [1-2].

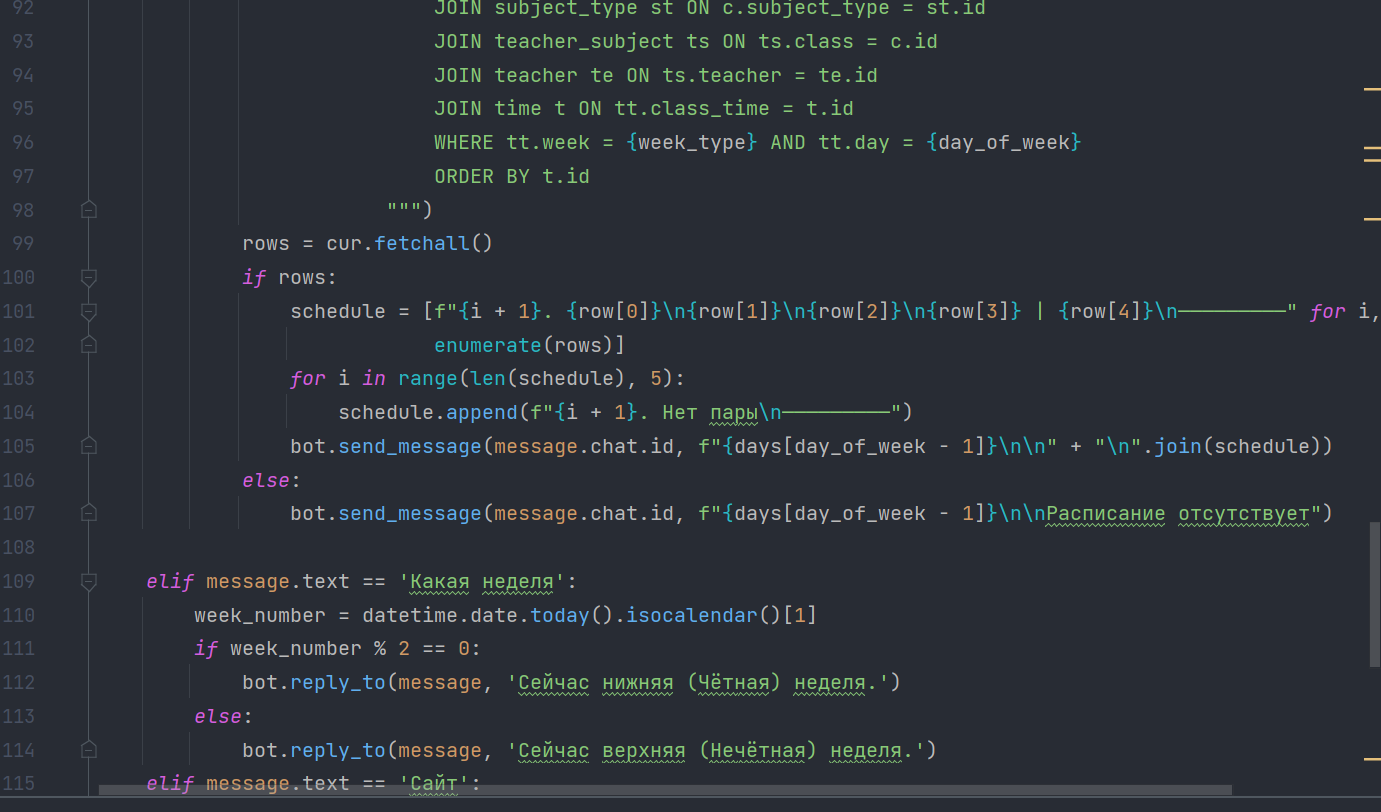
**Практическая реализация**

****

****

****

****

****

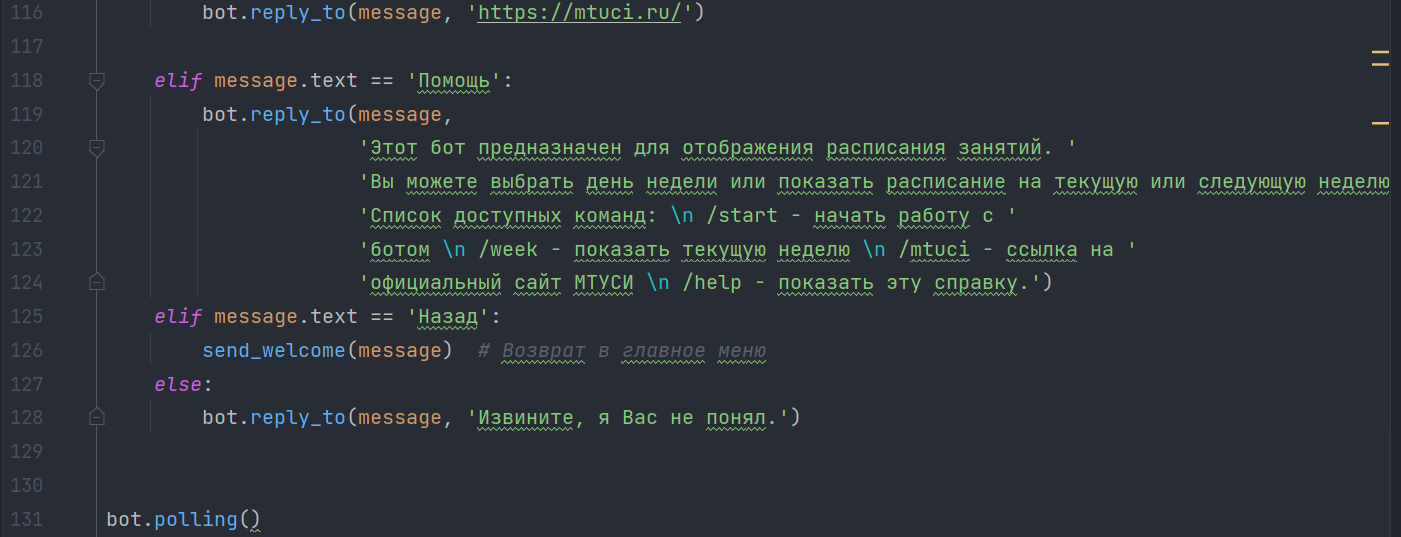
****

Рис. 1 – код бота

Данный код является программой-ботом для Telegram, который предназначен для отображения расписания занятий. Рассмотрим основные части кода:

1. Импорт необходимых модулей:

- `telebot` - модуль для работы с Telegram API.

- `psycopg2` - модуль для работы с базой данных PostgreSQL.

- `datetime` - модуль для работы с датой и временем.

2. Определение токена бота и создание экземпляра бота:

```

TOKEN = '6240479093:AAHQt\_UjSYTiphgQHTKMtdE79fNhjDdWfSc'

bot = telebot.TeleBot(TOKEN)

```

Вместо `'TOKEN'` должен быть указан реальный токен вашего Telegram бота.

3. Подключение к базе данных PostgreSQL:

```

con = psycopg2.connect(

database="lab7\_1",

user="postgres",

password="12345",

host="localhost",

port="5432"

)

```

Здесь указаны параметры подключения к базе данных: имя базы данных, имя пользователя, пароль, хост и порт.

4. Создание курсора для выполнения SQL-запросов:

```

cur = con.cursor()

```

5. Функция `get\_current\_week\_type()`:

Эта функция используется для определения текущей недели (верхняя или нижняя) на основе текущей даты. Она использует модуль `datetime` для получения текущей даты и `isocalendar()` для получения номера текущей недели. Результатом функции является число 1 или 2, которое представляет тип текущей недели.

6. Обработчик команды `/start`:

```

@bot.message\_handler(commands=['start'])

def send\_welcome(message):

# ...

```

Этот обработчик вызывается, когда пользователь отправляет команду `/start`. Он создает разметку клавиатуры с кнопками "Какая неделя", "Сайт", "Помощь" и "Расписание". Клавиатура отправляется пользователю вместе с сообщением "Выберите действие".

7. Обработчик всех сообщений:

```

@bot.message\_handler(func=lambda message: True)

def handle\_all\_message(message):

# ...

```

Этот обработчик вызывается для всех сообщений, не только команд. Он содержит множество условий для обработки различных текстовых сообщений, которые может отправить пользователь.

- Если сообщение пользователя равно "Расписание", то создается клавиатура с кнопками для выбора дня недели и отправляется пользователю.

- Если сообщение пользователя является названием дня недели (например, "Понедельник"), то выполняется SQL-запрос к базе данных для получения рас

писания занятий на указанный день. Результаты запроса отправляются пользователю в виде сообщения.

- Если сообщение пользователя равно "Какая неделя", то определяется тип текущей недели и отправляется соответствующее сообщение.

- Если сообщение пользователя равно "Сайт", то отправляется ссылка на официальный сайт МТУСИ.

- Если сообщение пользователя равно "Помощь", то отправляется справочное сообщение с описанием доступных команд.

- Если сообщение пользователя равно "Назад", то вызывается функция `send\_welcome()`, которая возвращает пользователя в главное меню.

8. Запуск бота:

```

bot.polling()

```

Эта функция начинает бесконечный цикл для получения и обработки новых сообщений от пользователей.

Это основная логика кода, которая обрабатывает команды пользователя и получает информацию о расписании занятий из базы данных для отправки обратно пользователю.

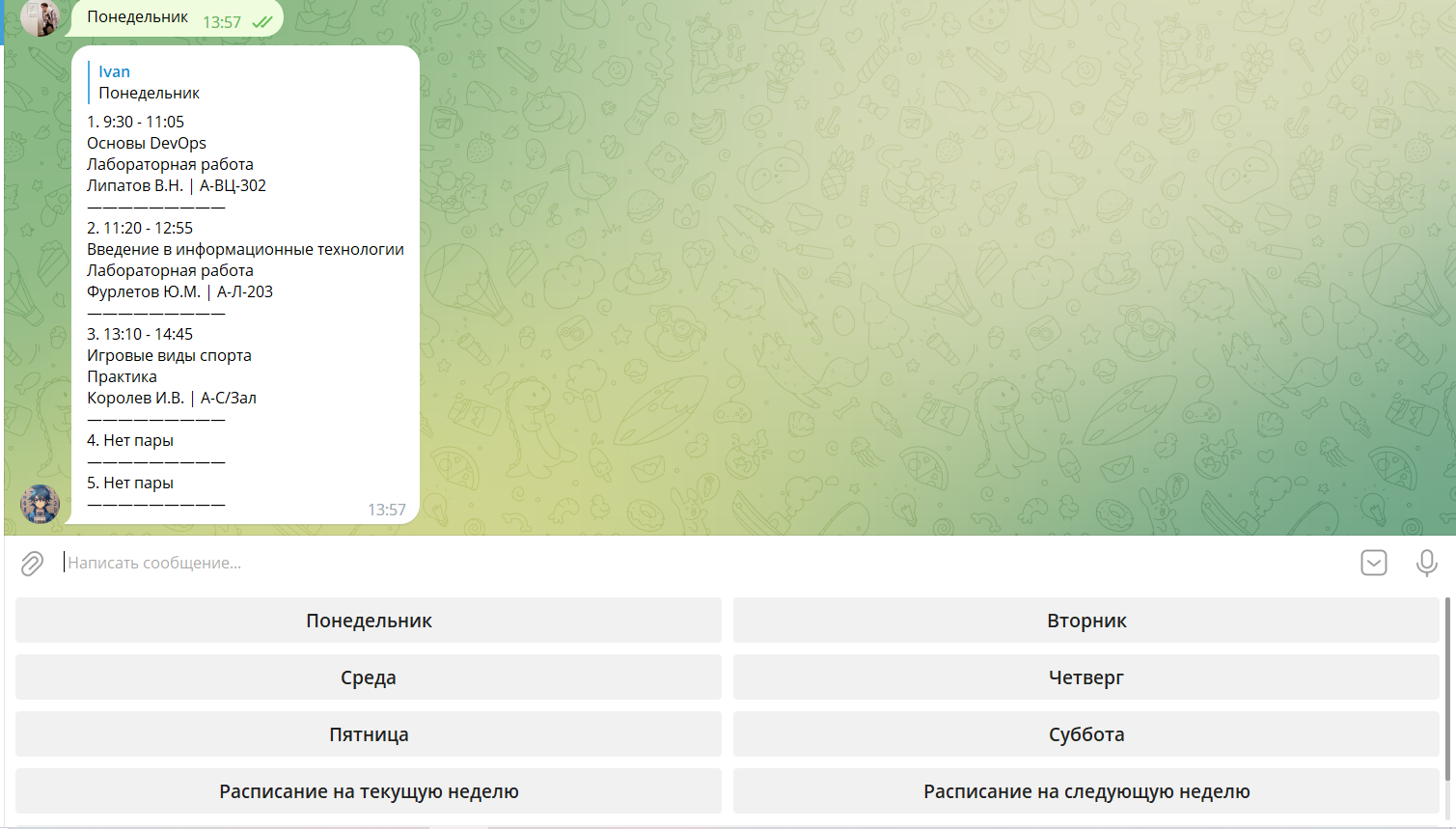
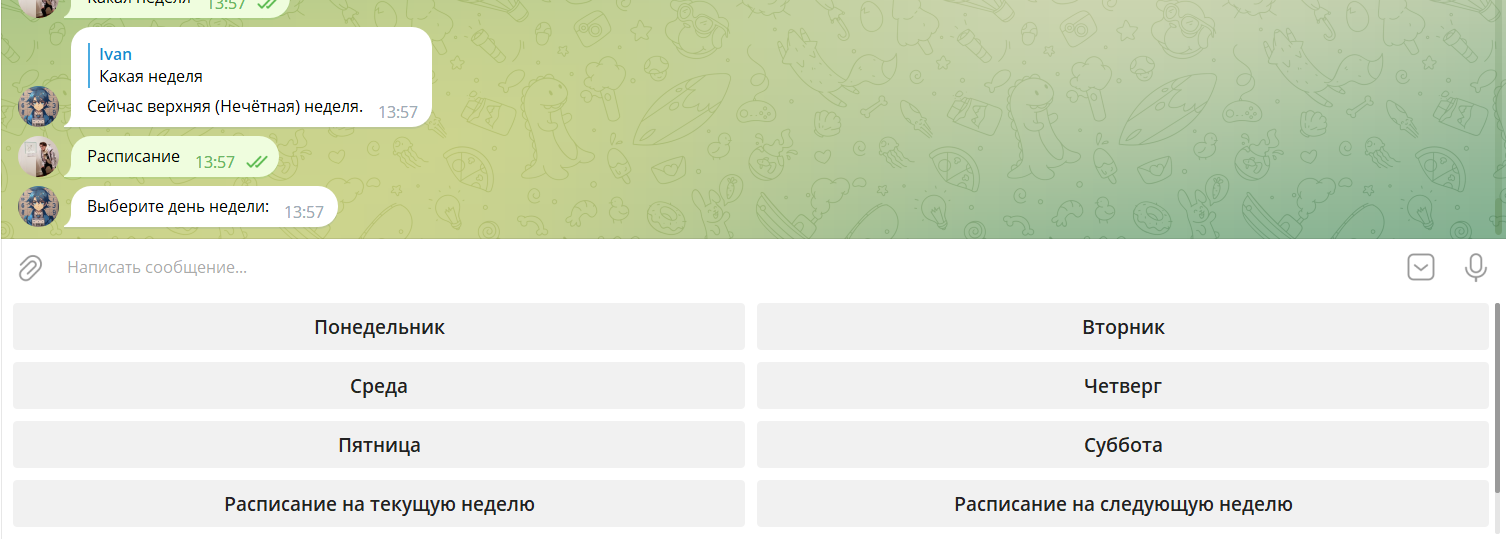
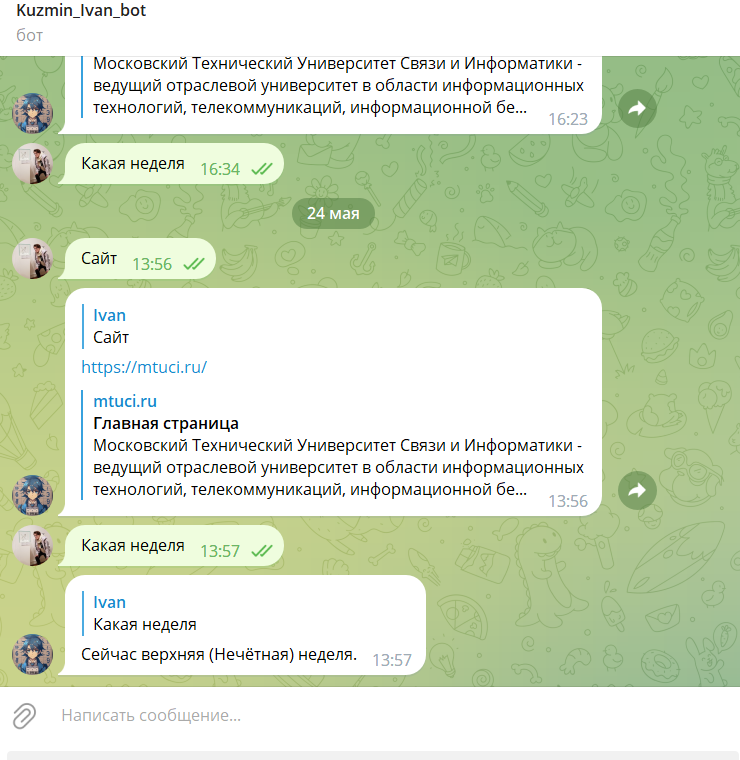
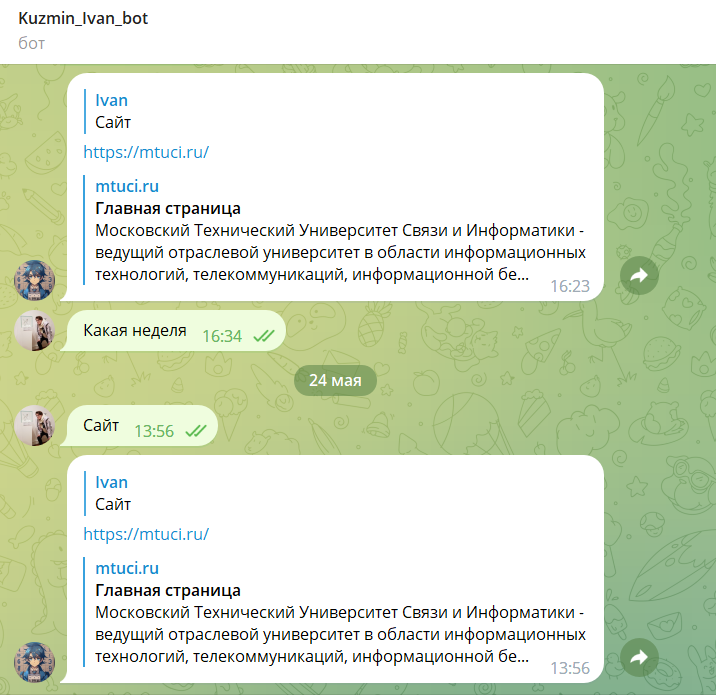
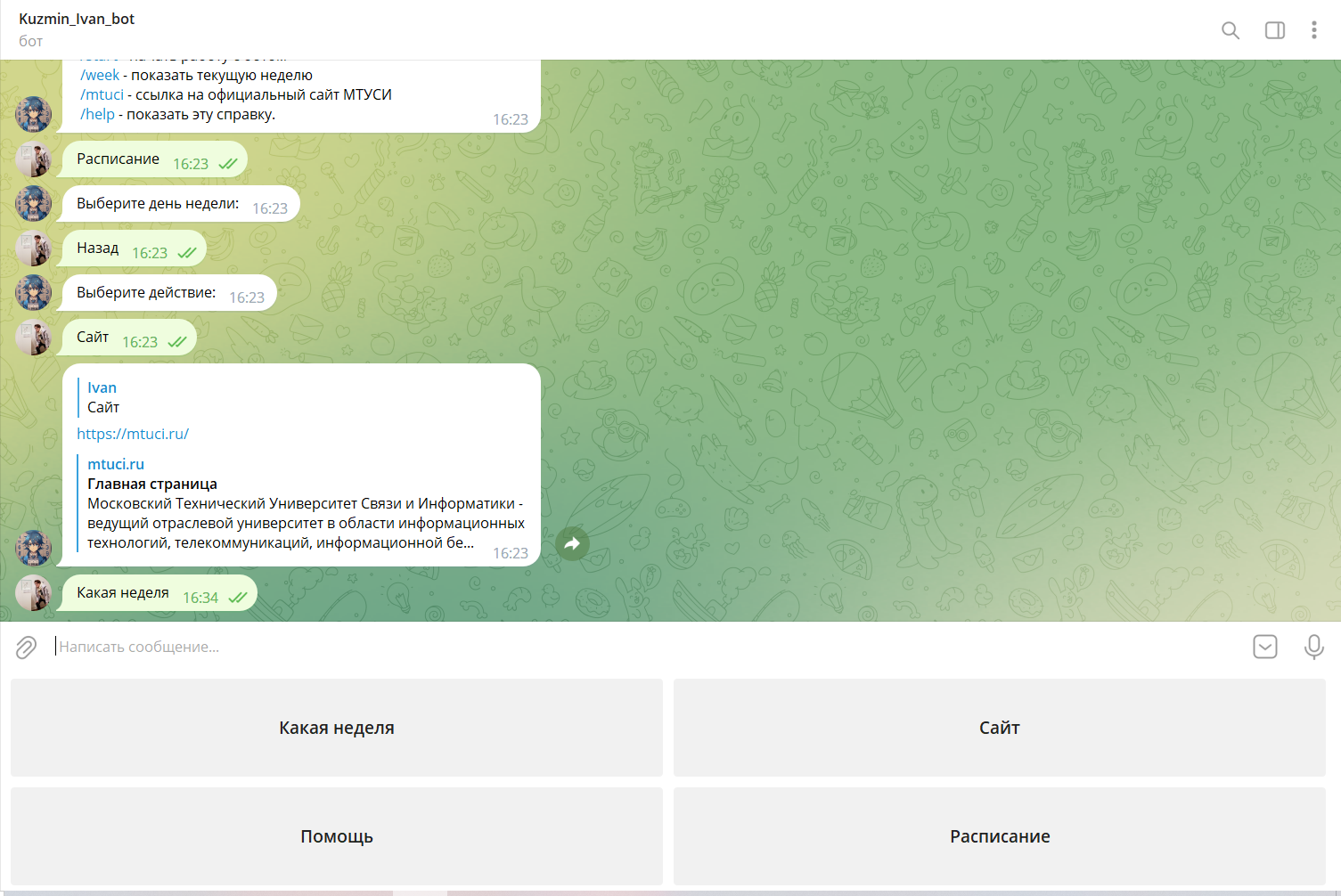


Рис. 2 – работа бота

# Заключение

# Список использованных источников