**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**

Кафедра «Информатики»

Практическая работа №7 по дисциплине ВвИТ

«Телебот»

Выполнил студент группы БИН2003 Авилов Д.Д.

Проверил: Аршинов Е.А.

Москва 2021

**Цель работы**

Создать своего бота в телеграмме, который будет выводить актуально расписание.

**Выполнение работы**

Код main.py:

import telebot

import psycopg2

import datetime

from telebot import types

token = "5098035151:AAGZtz9-tNEfDSdXK4MdSu98rVkNp91pv8Y"

conn = psycopg2.connect(database='postgres',

                        user='postgres',

                        password='',

                        host='localhost',

                        port='5432')

cursor = conn.cursor()

bot = telebot.TeleBot(token)

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

def start(message):

    keyboard = telebot.types.ReplyKeyboardMarkup(True)

    keyboard.row('Понедельник', 'Вторник', 'Среда', 'Четверг', 'Пятница')

    keyboard.row('ЧЕТ','НЕЧЕТ','сегодня','завтра','помощь')

    bot.send\_message(message.chat.id, 'Добрый день, чтобы узнать расписание напиши чет или нечет и интересующий день недели.', reply\_markup=keyboard)

def Monday(message):

    global week, cursor, bot

    if week == True:

        cursor.execute("SELECT \* FROM monc;")

    elif week == False:

        cursor.execute("SELECT \* FROM monn;")

    records = cursor.fetchall()

    result = ''

    for row in records:

        result+= row[1] + ' '

        result+= row[2] + ' '

        result+= row[3] + ' \n'

    bot.send\_message(message.chat.id, result)

def Tuesday(message):

    global week, cursor, bot

    if week == True:

        cursor.execute("SELECT \* FROM tuec;")

    elif week == False:

        cursor.execute("SELECT \* FROM tuen;")

    records = cursor.fetchall()

    result = ''

    for row in records:

        result+= row[1] + ' '

        result+= row[2] + ' \n'

    bot.send\_message(message.chat.id, result)

def Wednesday(message):

    global week, cursor, bot

    if week == True:

        cursor.execute("SELECT \* FROM wenc;")

    elif week == False:

        cursor.execute("SELECT \* FROM wenn;")

    records = cursor.fetchall()

    result = ''

    for row in records:

        result+= row[1] + ' '

        result+= row[2] + ' \n'

    bot.send\_message(message.chat.id, result)

def Thursday(message):

    global week, cursor, bot

    if week == True:

        cursor.execute("SELECT \* FROM thurc;")

    elif week == False:

        cursor.execute("SELECT \* FROM thurn;")

    records = cursor.fetchall()

    result = ''

    for row in records:

        result+= row[1] + ' '

        result+= row[2] + ' \n'

    bot.send\_message(message.chat.id, result)

def Friday(message):

    global week, cursor, bot

    if week == True:

        cursor.execute("SELECT \* FROM fric;")

    elif week == False:

        cursor.execute("SELECT \* FROM frin;")

    records = cursor.fetchall()

    result = ''

    for row in records:

        result+= row[1] + ' '

        result+= row[2] + ' \n'

    bot.send\_message(message.chat.id, result)

def next(bot, message, day):

    if day == 1:

        Monday(message)

    if day == 2:

        Tuesday(message)

    if day == 3:

        Wednesday(message)

    if day == 4:

        Thursday(message)

    if day == 5:

        Friday(message)

    if day == 6:

        bot.send\_message(message.chat.id, 'В субботу нет пар.')

    if day == 7:

        bot.send\_message(message.chat.id, 'В воскресенье нет пар.')

    if day == 8:

        Monday(message)

@bot.message\_handler(content\_types=['text'])

def response(message):

    global week

    if message.text.lower() == 'чет':

        week = True

        bot.send\_message(message.chat.id, 'Вывожу расписание для четной недели')

    elif message.text.lower() == 'нечет':

        week = False

        bot.send\_message(message.chat.id, 'Вывожу расписание для нечетной недели')

    if message.text.lower() == 'понедельник':

        Monday(message)

    if message.text.lower() == 'вторник':

        Tuesday(message)

    if message.text.lower() == 'среда':

        Wednesday(message)

    if message.text.lower() == 'четверг':

        Thursday(message)

    if message.text.lower() == 'пятница':

        Friday(message)

    if message.text.lower() == 'сегодня':

        day = int(datetime.datetime.now().strftime('%w'))

        next(bot, message, day)

    if message.text.lower() == 'завтра':

        day = int(datetime.datetime.now().strftime('%w')) + 1

        next(bot, message, day)

    if message.text.lower() == 'помощь':

        bot.send\_message(message.chat.id, 'Чтобы вывести расписание, вам необходимо выбрать четность недели')

bot.infinity\_polling()

Пример работы приложения представлен ниже:

Изображение выглядит как текст, электроника

Автоматически созданное описание

Рисунок 1 – Начальный этап работы

Изображение выглядит как текст, устройство, счетчик, закрыть

Автоматически созданное описание

Рисунок 2 – Исключения в ходе работы программы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 – Дополнительные функции программы

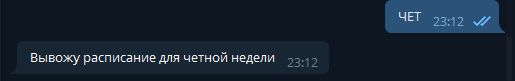


Рисунок 4 – Выбор четности недели

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 – Расписание на понедельник

Изображение выглядит как текст, снимок экрана, монитор, черный

Автоматически созданное описание

Рисунок 6 – Полный функционал приложения

**Вывод:**

Бот запускается и выводит расписание.