**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 Филимонов И.А.

Проверил: Аршинов Е. А.

Москва 2021

**1. Цель работы:**

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

**2. Выполнение работы:**

Код исполняемого файла main.py:

import telebot

import psycopg2

import datetime

from telebot import types

token = "2118799805:AAFJYHALKcDFQI4a\_tEBtW1HE29FZRaeSwQ"

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

user='oleg',

password='oleg',

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')

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[0] + ' '

result+= row[1] + ' '

result+= row[2] + ' \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[0] + ' '

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 wedC;")

elif week == False:

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

records = cursor.fetchall()

result = ''

for row in records:

result+= row[0] + ' '

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 thuC;")

elif week == False:

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

records = cursor.fetchall()

result = ''

for row in records:

result+= row[0] + ' '

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 fricC;")

elif week == False:

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

records = cursor.fetchall()

result = ''

for row in records:

result+= row[0] + ' '

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)

bot.infinity\_polling()

Конечный результат:



Рисунок 1

**3. Вывод:**

В итоге мы достигли цели, бот успешно запустился и выводит расписание.