**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

«Телеграм-бот»

Выполнил студент группы БИН2003 Беляков А.Н.

Проверил: Егор Аршинов

Москва 2021

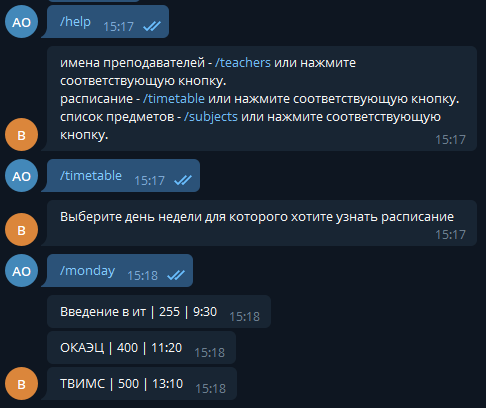
**1. Цель работы:**

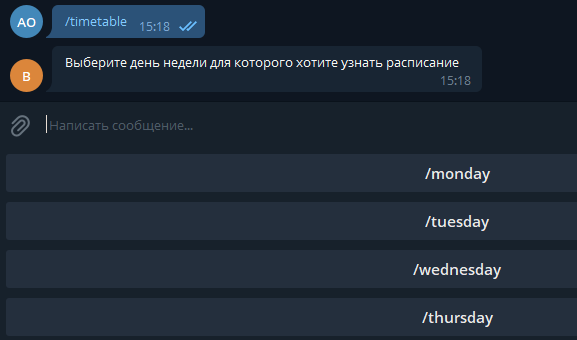
Создать своего телеграм-бота с расписанием занятий

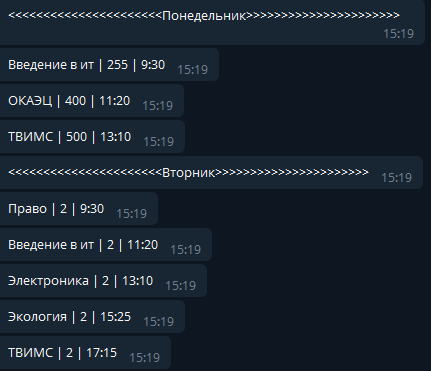
**2. Выполнение работы:**

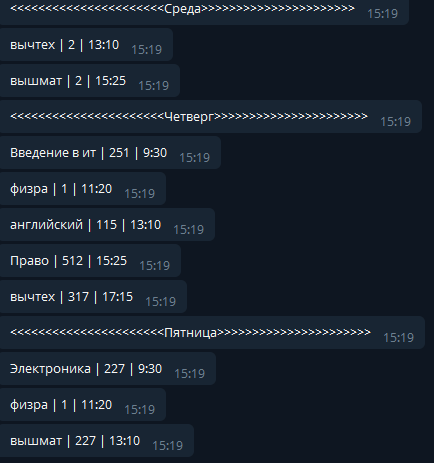
import telebot  
from telebot import types  
import psycopg2  
from datetime import date  
  
bot = telebot.TeleBot('2135105967:AAHWydKiURg9NU6z44q52WlnlLnv1ZZq\_30')  
  
conn = psycopg2.connect(database="timetable",  
 user="postgres",  
 password="1488",  
 host="localhost",  
 port="5432")  
  
def get\_week\_num():  
 first\_day = date(2021, 8, 30)  
 today = date.today()  
 delta = (today - first\_day).days  
 week\_number = (delta // 7) + 1  
 return week\_number  
  
def execute\_read\_query(connection, query):  
 cursor = connection.cursor()  
 result = None  
 cursor.execute(query)  
 result = cursor.fetchall()  
 return result  
  
  
@bot.message\_handler(commands=['start', 'back'])  
def start\_message(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/help')  
 btn2 = types.KeyboardButton('/teachers')  
 btn3 = types.KeyboardButton('/timetable')  
 btn4 = types.KeyboardButton('/subjects')  
 markup.add(btn1)  
 markup.add(btn2)  
 markup.add(btn3)  
 markup.add(btn4)  
 bot.send\_message(message.chat.id, 'Для того, чтобы узнать все команды пропишите команду /help '  
 'или нажмите на кнопку Help', reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['help'])  
def helper(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/teachers')  
 btn2 = types.KeyboardButton('/timetable')  
 btn3 = types.KeyboardButton('/subjects')  
 markup.add(btn1)  
 markup.add(btn2)  
 markup.add(btn3)  
 bot.send\_message(message.chat.id, 'имена преподавателей - '  
 '/teachers или нажмите соответствующую кнопку.\n'  
 'расписание - '  
 '/timetable или нажмите соответствующую кнопку.\n'  
 'список предметов - '  
 '/subjects или нажмите соответствующую кнопку.', reply\_markup=markup)  
  
@bot.message\_handler(commands=['subjects'])  
def sub(message):  
 text = "SELECT \* FROM lst.subject"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, str(subjects))  
  
  
@bot.message\_handler(commands=['teachers'])  
def teach(message):  
 text = "SELECT \* FROM lst.teacher"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, str(subjects))  
  
  
@bot.message\_handler(commands=['timetable'])  
def timetable(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/monday')  
 btn2 = types.KeyboardButton('/tuesday')  
 btn3 = types.KeyboardButton('/wednesday')  
 btn4 = types.KeyboardButton('/thursday')  
 btn5 = types.KeyboardButton('/friday')  
 btn6 = types.KeyboardButton('/chet')  
 btn7 = types.KeyboardButton('/nechet')  
 markup.add(btn1)  
 markup.add(btn2)  
 markup.add(btn3)  
 markup.add(btn4)  
 markup.add(btn5)  
 markup.add(btn6)  
 markup.add(btn7)  
 bot.send\_message(message.chat.id, 'Выберите день недели для которого хотите узнать расписание', reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['monday'])  
def mon(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/back')  
 markup.add(btn1)  
 week\_num = '2' if get\_week\_num()%2 == 1 else ''  
 text = f"SELECT subject, room\_numb, start\_time FROM lst.timetable{week\_num} WHERE day = 'monday'"  
 subjects = execute\_read\_query(conn, text)  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv, reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['tuesday'])  
def tue(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/back')  
 markup.add(btn1)  
 week\_num = '2' if get\_week\_num() % 2 == 1 else ''  
 text = f"SELECT subject, room\_numb, start\_time FROM lst.timetable{week\_num} WHERE day = 'tuesday'"  
 subjects = execute\_read\_query(conn, text)  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv, reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['wednesday'])  
def wed(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/back')  
 markup.add(btn1)  
 week\_num = '2' if get\_week\_num() % 2 == 1 else ''  
 text = f"SELECT subject, room\_numb, start\_time FROM lst.timetable{week\_num} WHERE day = 'wednesday'"  
 subjects = execute\_read\_query(conn, text)  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv, reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['thursday'])  
def thu(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/back')  
 week\_num = '2' if get\_week\_num() % 2 == 1 else ''  
 markup.add(btn1)  
 text = f"SELECT subject, room\_numb, start\_time FROM lst.timetable{week\_num} WHERE day = 'thursday'"  
 subjects = execute\_read\_query(conn, text)  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv, reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['friday'])  
def fri(message):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)  
 btn1 = types.KeyboardButton('/back')  
 markup.add(btn1)  
 week\_num = '2' if get\_week\_num() % 2 == 1 else ''  
 text = f"SELECT subject, room\_numb, start\_time FROM lst.timetable{week\_num} WHERE day = 'friday'"  
 subjects = execute\_read\_query(conn, text)  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv, reply\_markup=markup)  
  
  
@bot.message\_handler(commands=['chet'])  
def chet(message):  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable where day = 'monday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Понедельник>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable where day = 'tuesday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Вторник>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable where day = 'wednesday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Среда>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable where day = 'thursday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Четверг>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable where day = 'friday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Пятница>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
  
 ########################  
  
@bot.message\_handler(commands=['nechet'])  
def nechet(message):  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable2 where day = 'monday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Понедельник>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable2 where day = 'tuesday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Вторник>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable2 where day = 'wednesday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Среда>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable2 where day = 'thursday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Четверг>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
 text = "SELECT subject, room\_numb, start\_time FROM lst.timetable2 where day = 'friday'"  
 subjects = execute\_read\_query(conn, text)  
 bot.send\_message(message.chat.id, '<<<<<<<<<<<<<<<<<<<<<<Пятница>>>>>>>>>>>>>>>>>>>>>>')  
 for row in subjects:  
 otv = ''  
 otv = row[0] + ' | ' + row[1] + ' | ' + row[2]  
 bot.send\_message(message.chat.id, otv)  
  
bot.infinity\_polling()

**3. Результат работы:**









**4. Вывод: Мы написали телеграм-бота с расписанием занятий.**