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»

Лабораторная работа №4

«Бот-рассписание»

По дисциплине

«Введение в ИТ»

Выполнил: студент

Группы БИН2002

Сас А.М.

Проверил: Чайка А.Д.

Москва 2021 г.

Содержание

[**1.** **Цель работы** 3](#_Toc89954306)

[**2.** **Выполнение** 3](#_Toc89954307)

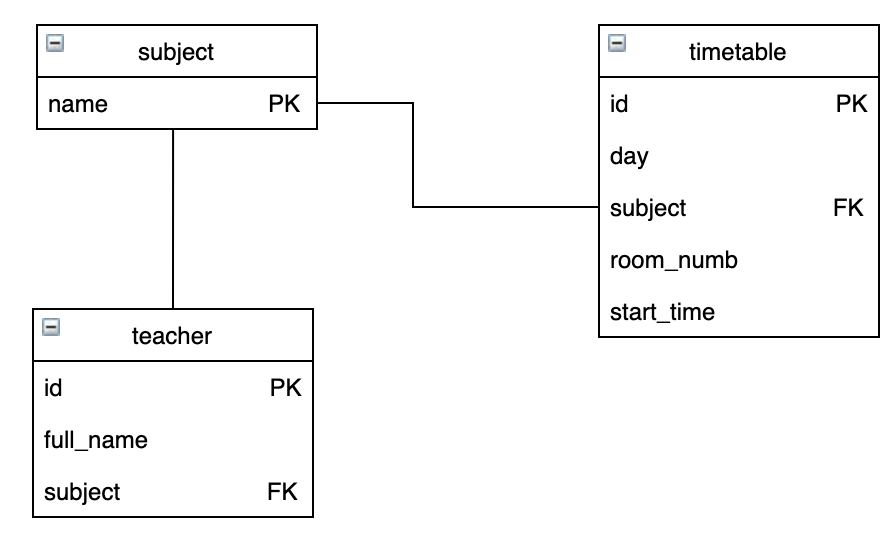
[**3.** **Вывод** 7](#_Toc89954308)

1. **Цель работы**

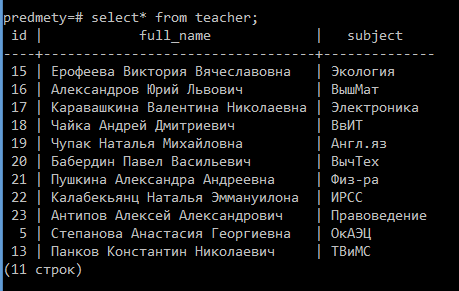
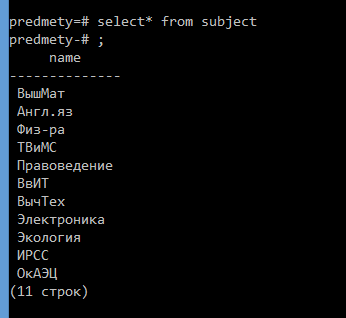
Создать оконное приложение, позволяющее редактировать базу данных с расписанием Вашей группы.

1. **Выполнение**

Для выполнения данной лабораторной работы, требуется создание sql базы данных с помощью программы PostgreSQL, включающей в себя три таблиц по схеме:



После создания база дынных выглядит так (Рисунок 1)



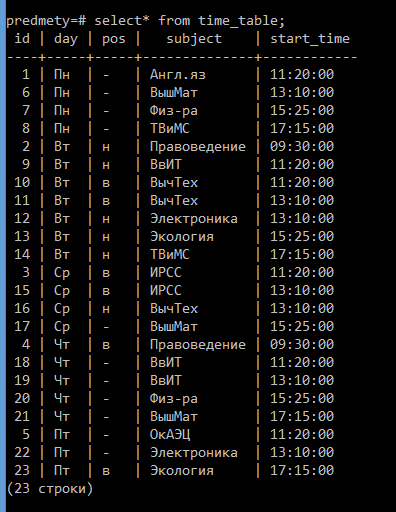


Рисунок 1 – Данные из sql таблиц

Когда пользователь переходит к созданному телеграм-боту(@nastyasas\_timetable\_bot) и вводит команду /start, ему показывается приветствие и выводятся кнопки ‘Расписание на сегодня’ ‘Расписание на завтра’ ‘Расписание’, при нажатии на кнопки выводится информация о расписании (Рисунок 2)

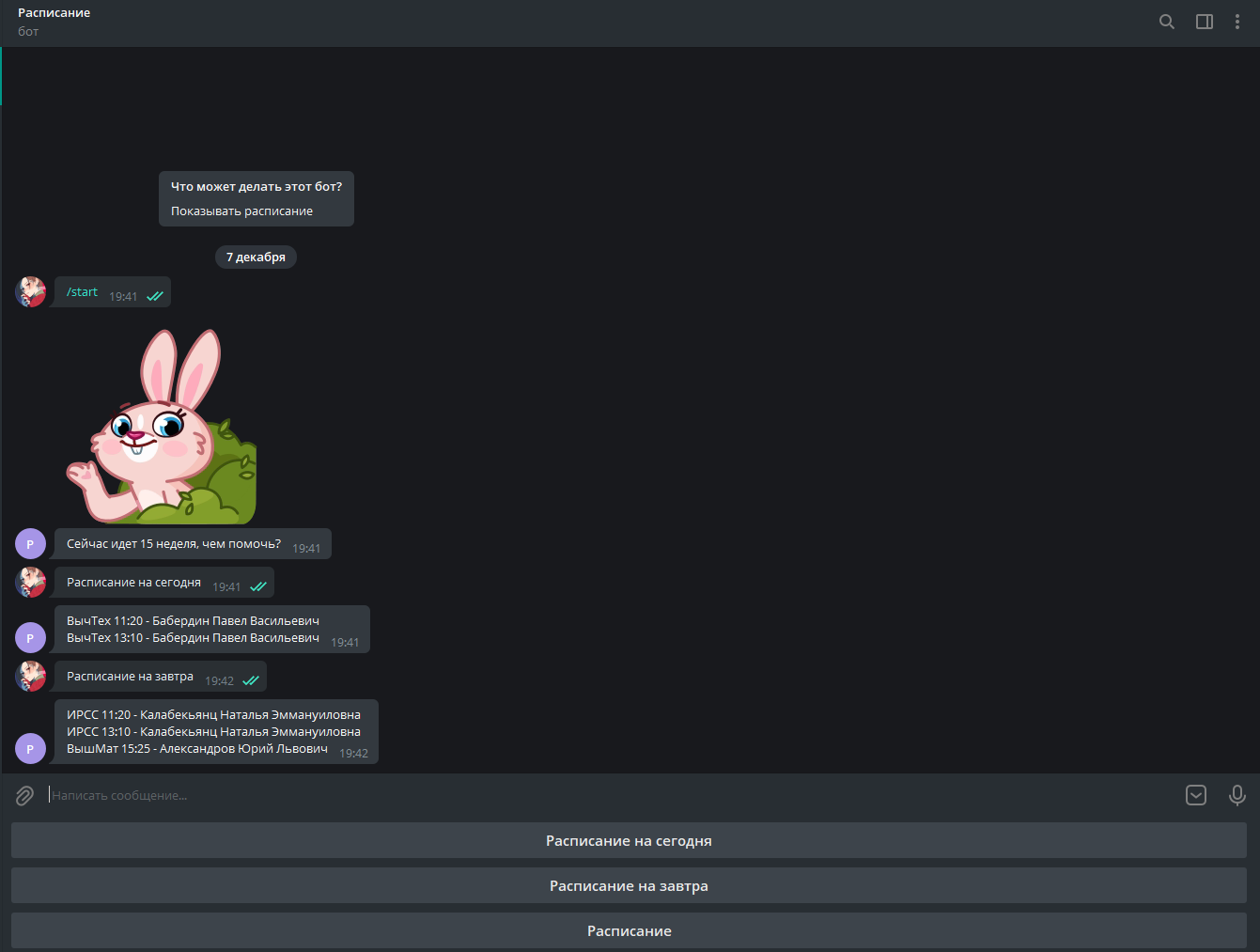


Рисунок 2 - Информация о расписании

При нажатии на кнопку расписание пользователь выводится 5 кнопок с выбором дня недели и кнопка ‘Назад’ возвращающая пользователя обратно, при нажатии на одну из 5 кнопок выводится расписание на выбранный день (Рисунок 3)

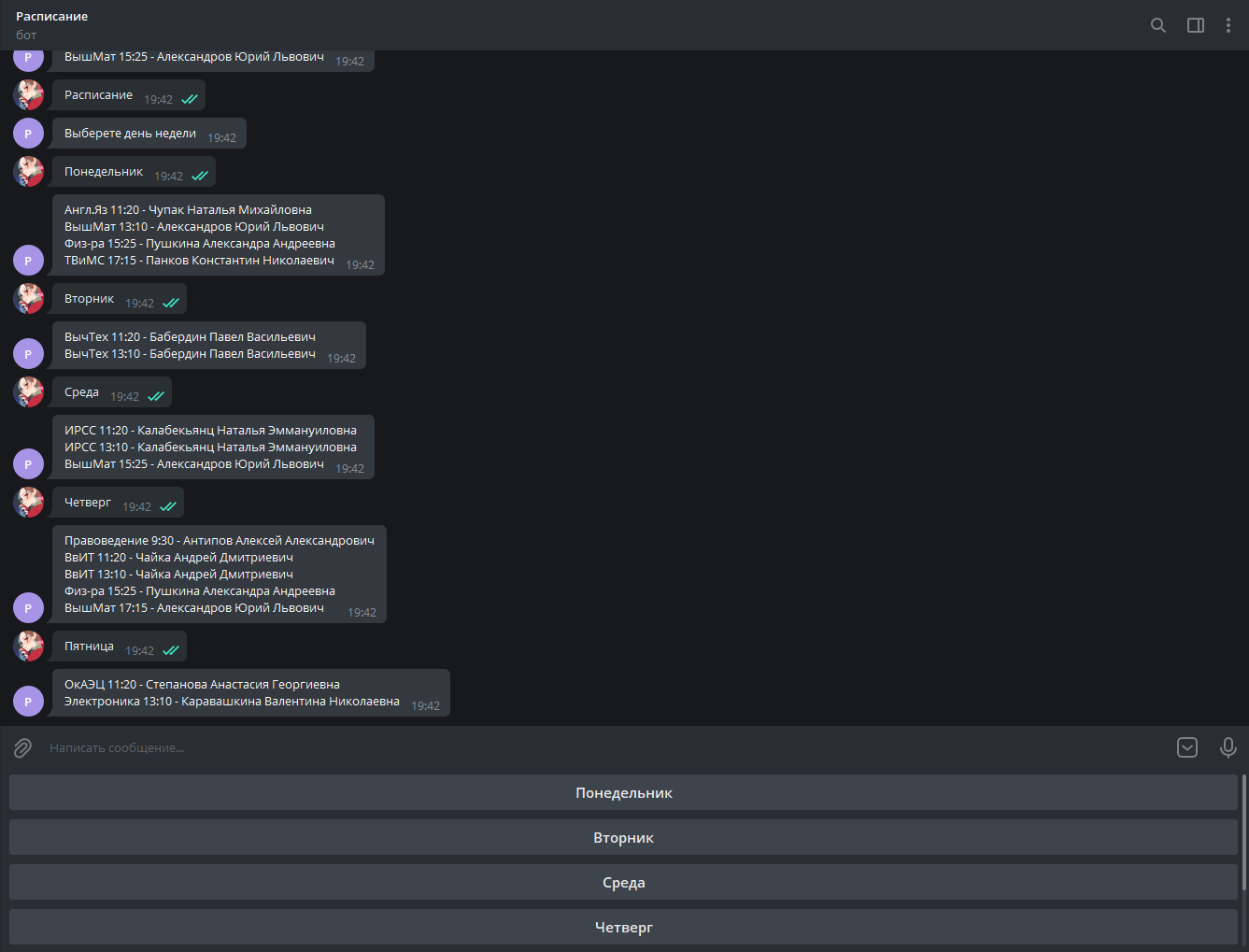


Рисунок 3 - Расписание

1. **Вывод**

В ходе выполнения работы, я научилась созданию телеграм-ботов с использованием pyTelegramBotAPI и PostgreSQL

**Приложение:**

**Программный код файла main.py**

import datetime

import psycopg2

import telebot

from telebot import types

from math import ceil

bot = telebot.TeleBot('2111798061:AAGSJfnEvHgInIK-h51lv9T\_Fbj4522413A')

conn = psycopg2.connect(database="predmety", user="postgres", password="12345", host="localhost", port="5432")

cursor = conn.cursor()

day = ['Пн', 'Вт', 'Ср', 'Чт', 'Пт', 'Субботу', 'Воскресенье']

if datetime.datetime.now().month > 8:

d1 = datetime.datetime.strptime("01-09-" + str(datetime.datetime.now().year), "%d-%m-%Y")

else:

d1 = datetime.datetime.strptime("31-01-" + str(datetime.datetime.now().year), "%d-%m-%Y")

d2 = datetime.datetime.strptime(str(datetime.datetime.date(datetime.datetime.now())), "%Y-%m-%d")

if datetime.datetime.isoweekday(d1) < 5:

week\_n = ceil(((d2 - d1).days + datetime.datetime.weekday(d1)) / 7)

elif datetime.datetime.isoweekday(d1) == 6:

week\_n = ceil(((d2 - d1).days - 2) / 7)

elif datetime.datetime.isoweekday(d1) == 7:

week\_n = ceil(((d2 - d1).days - 1) / 7)

def menu(message):

markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)

btn1 = types.KeyboardButton('Расписание на сегодня')

btn2 = types.KeyboardButton('Расписание на завтра')

btn3 = types.KeyboardButton('Расписание')

markup.add(btn1)

markup.add(btn2)

markup.add(btn3)

bot.send\_message(message.chat.id, "Сейчас идет " + str(week\_n) + " неделя, чем помочь?", reply\_markup=markup)

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

def start\_message(message):

bot.send\_sticker(message.chat.id, 'CAACAgIAAxkBAAEDUq1hmB7iUFjM4IJQoSMfqfR4Jt\_OxwACbgUAAj-VzAqGOtldiLy3NSIE')

menu(message)

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

def help\_message(message):

bot.send\_message(message.chat.id, "Я показываю расписание группы БИН2002!")

bot.send\_sticker(message.chat.id, 'CAACAgIAAxkBAAEDUrFhmCBC6oQw\_C0J0pADVtzYNBvlKwACRgEAAiI3jgQiieP6A4eLHiIE')

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

def reply\_message(message):

def time\_table(a, b):

cursor.execute(

"select time\_table.subject, time\_table.start\_time, teacher.full\_name from time\_table join teacher on "

"teacher.subject = time\_table.subject where day = '" + a + "' and (pos = '-' or pos = '" + b + "') "

"order by time\_table.start\_time;")

row = list(cursor.fetchall())

mess = ''

for i in row:

mess += str(i[0]) + ' '

mess += str(i[1]) + ' - '

mess += str(i[2]) + '\n'

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

def week\_pos(a):

if week\_n % 2 > 0:

time\_table(a, 'в')

else:

time\_table(a, 'н')

if message.text == "Расписание на сегодня":

if datetime.datetime.weekday(datetime.datetime.now()) < 5:

week\_pos(day[datetime.datetime.weekday(datetime.datetime.now())])

else:

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

"В " + day[datetime.datetime.weekday(datetime.datetime.now())] + " не учимся!")

bot.send\_sticker(message.chat.id,

'CAACAgIAAxkBAAEDUrVhmCHu7coC0T2qWDhIejufAAFyc-UAAj8BAAIiN44ENDnV16oKRgEiBA')

if message.text == "Расписание на завтра":

if datetime.datetime.weekday(datetime.datetime.now()) < 4:

week\_pos(day[datetime.datetime.weekday(datetime.datetime.now()) + 1])

else:

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

"В " + (day[datetime.datetime.weekday(datetime.datetime.now()) + 1]) + " не учимся!")

bot.send\_sticker(message.chat.id,

'CAACAgIAAxkBAAEDUrVhmCHu7coC0T2qWDhIejufAAFyc-UAAj8BAAIiN44ENDnV16oKRgEiBA')

if message.text == "Расписание":

markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)

btn1 = types.KeyboardButton('Понедельник')

btn2 = types.KeyboardButton('Вторник')

btn3 = types.KeyboardButton('Среда')

btn4 = types.KeyboardButton('Четверг')

btn5 = types.KeyboardButton('Пятница')

btn6 = types.KeyboardButton('Назад')

markup.add(btn1)

markup.add(btn2)

markup.add(btn3)

markup.add(btn4)

markup.add(btn5)

markup.add(btn6)

bot.send\_message(message.chat.id, "Выберете день недели", reply\_markup=markup)

if message.text == "Понедельник":

week\_pos(day[0])

if message.text == "Вторник":

week\_pos(day[1])

if message.text == "Среда":

week\_pos(day[2])

if message.text == "Четверг":

week\_pos(day[3])

if message.text == "Пятница":

week\_pos(day[4])

if message.text == "Назад":

menu(message)

bot.send\_sticker(message.chat.id, 'CAACAgIAAxkBAAEDUrNhmCDfA1kbsHEzMPpeDwE55iTR2wACZgIAAladvQpDYzS\_ujiqhCIE')

bot.infinity\_polling()