

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Радиотехнический»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №3

«Разработка простого бота для Telegram с использованием
языка Python»

Выполнил:
Студент группы РТ5-31Б
Серик И. Н.
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.
Подпись и дата:

Москва, 2023 г.

Задание

Разработайте простого бота для Telegram. Бот должен использовать функциональность создания кнопок.

lab_bot.py

```
import telebot
from telebot import types

bot = telebot.TeleBot('6739358174:AAFD_EA-qXv1xgoikkDWoq6gmFuU-nlRCjQ')

# Словарь для хранения информации о лабораторных работах и ссылок
labs_info = {
    'lab1': {'text': 'Первая лабораторная работа выполнена в трех вариантах:\n- на\npython с применением процедурной парадигмы\n- на python с применением объектно-\nориентированной парадигмы\n- на языке C#', 'url':\n'https://github.com/serikivan/PCPL-course/blob/main/lab_python_intro/Report.pdf'},
    'lab2': {'text': 'Вторая лабораторная работа о функциональных возможностях языка\nPython', 'url': 'https://github.com/serikivan/PCPL-\ncourse/blob/main/lab_python_fp/Report.pdf'},
    'lab3': {'text': 'Третья лабораторная работа - этот чат бот', 'url':\n'https://github.com/serikivan/PCPL-course/blob/main/lab_bot1/Report.pdf'},
}

# Словарь для хранения информации о рубежных контролях и ссылок
rk_info = {
    'rk1': {'text': 'Отчет РК1', 'url': 'https://github.com/serikivan/PCPL-\ncourse/blob/main/RK1/rk1.pdf'},
    'rk2': {'text': 'Отчет РК2', 'url': 'https://github.com/serikivan/PCPL-\ncourse/blob/main/RK2/rk2.pdf'},
}

# Стек для хранения предыдущих состояний
state_stack = []

@bot.message_handler(commands=['start'])
def handle_start(message):
    show_main_menu(message)

def show_main_menu(message):
    keyboard = types.InlineKeyboardMarkup(row_width=2)
    repo_button = types.InlineKeyboardButton('Репозиторий',
url="https://github.com/serikivan/PCPL-course/wiki")
    labs_button = types.InlineKeyboardButton('Лабы', callback_data='labs')
    rk_button = types.InlineKeyboardButton('РК', callback_data='rk')
    keyboard.add(repo_button, labs_button, rk_button)
    bot.send_message(message.chat.id, 'Выберите интересующий раздел',
reply_markup=keyboard)

@bot.callback_query_handler(func=lambda call: True)
def handle_callback(call):
    if call.data == 'labs':
        show_labs_menu(call.message)
    elif call.data in labs_info:
        show_lab_info(call.message, labs_info[call.data])
    elif call.data == 'rk':
        show_rk_menu(call.message)
    elif call.data in rk_info:
        show_rk_info(call.message, rk_info[call.data])
    elif call.data == 'back':
        show_previous_menu(call.message)
    elif call.data == 'back_to_labs':
        show_labs_menu(call.message)
```

```

elif call.data == 'back_to_rk':
    show_rk_menu(call.message)

def show_labs_menu(message):
    labs_keyboard = types.InlineKeyboardMarkup(row_width=1)
    for lab_key in labs_info:
        labs_keyboard.add(types.InlineKeyboardButton(lab_key.capitalize(),
callback_data=lab_key))
    labs_keyboard.add(types.InlineKeyboardButton('Назад', callback_data='back'))
    state_stack.append(show_main_menu)
    bot.edit_message_text(chat_id=message.chat.id, message_id=message.message_id,
        text='Выберите лабораторную работу:',
reply_markup=labs_keyboard)

def show_lab_info(message, lab_info):
    keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton(text="Перейти", url=lab_info['url'])
    back_button = types.InlineKeyboardButton('Назад к лабам',
callback_data='back_to_labs')
    keyboard.add(url_button, back_button)
    bot.edit_message_text(chat_id=message.chat.id, message_id=message.message_id,
        text=lab_info['text'], reply_markup=keyboard)

def show_rk_menu(message):
    rk_keyboard = types.InlineKeyboardMarkup(row_width=1)
    for rk_key in rk_info:
        rk_keyboard.add(types.InlineKeyboardButton(rk_key.capitalize(),
callback_data=rk_key))
    rk_keyboard.add(types.InlineKeyboardButton('Назад', callback_data='back'))
    state_stack.append(show_main_menu)
    bot.edit_message_text(chat_id=message.chat.id, message_id=message.message_id,
        text='Выберите рубежный контроль:',
reply_markup=rk_keyboard)

def show_rk_info(message, rk_info):
    keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton(text="Перейти", url=rk_info['url'])
    back_button = types.InlineKeyboardButton('Назад к рубежным контролям',
callback_data='back_to_rk')
    keyboard.add(url_button, back_button)
    bot.edit_message_text(chat_id=message.chat.id, message_id=message.message_id,
        text=rk_info['text'], reply_markup=keyboard)

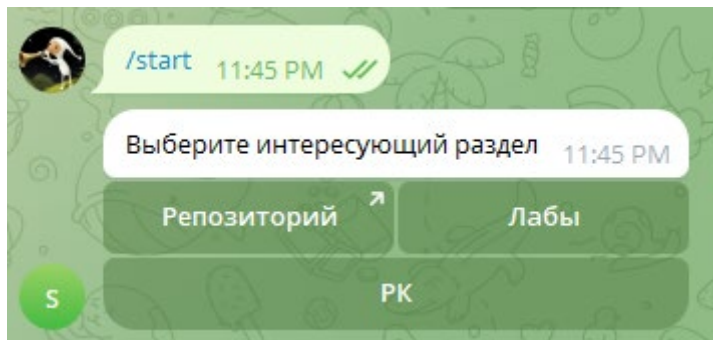
def show_previous_menu(message):
    if state_stack:
        previous_state = state_stack.pop()
        previous_state(message)

if __name__ == "__main__":
    bot.polling()

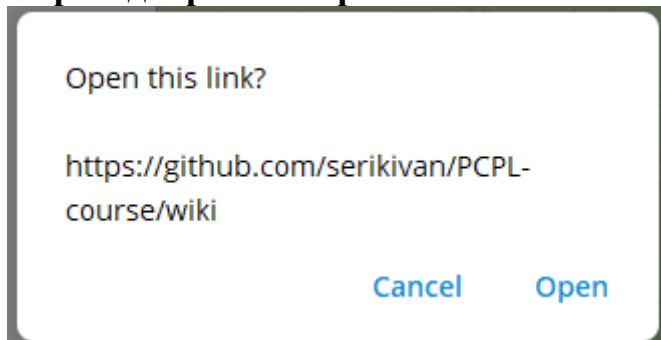
```

Результаты

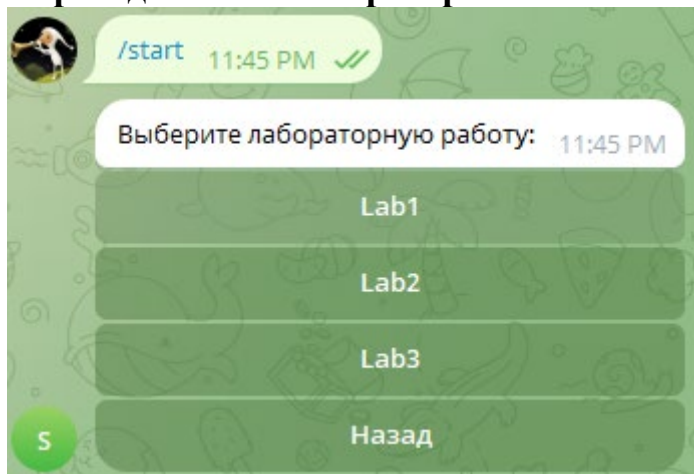
Главное меню



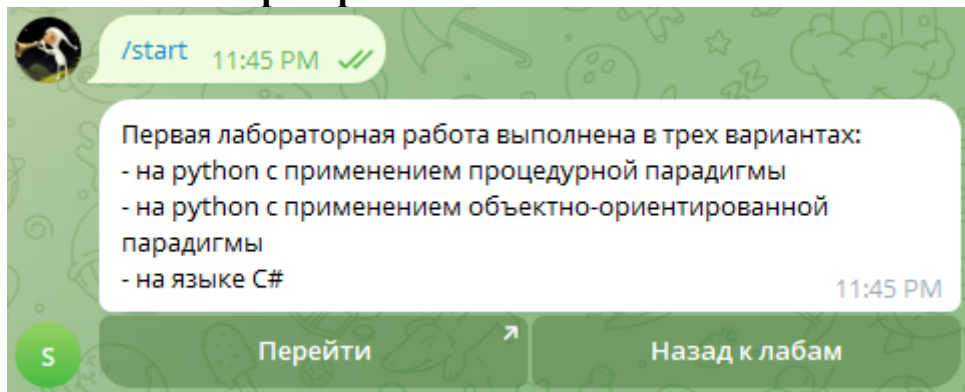
Переход в репозиторий

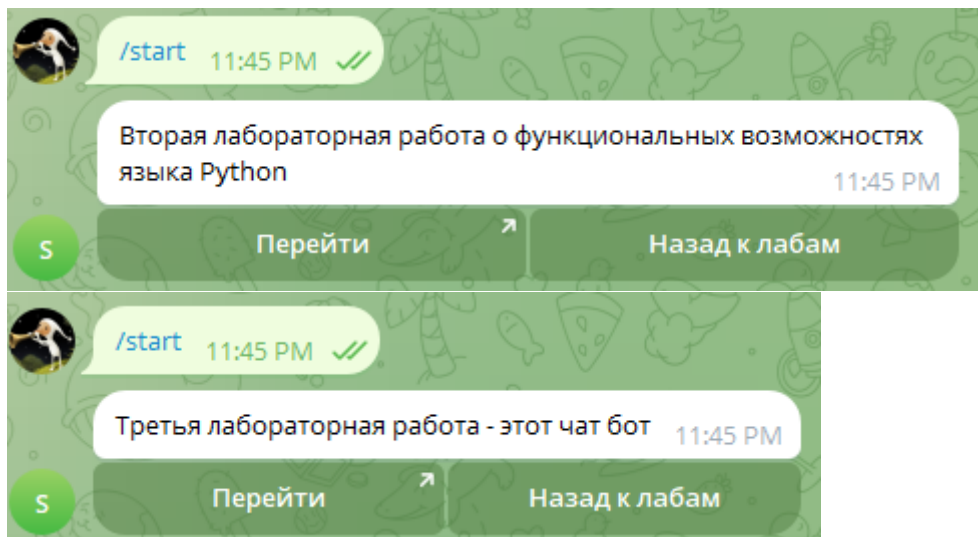


Переход к меню лабораторных

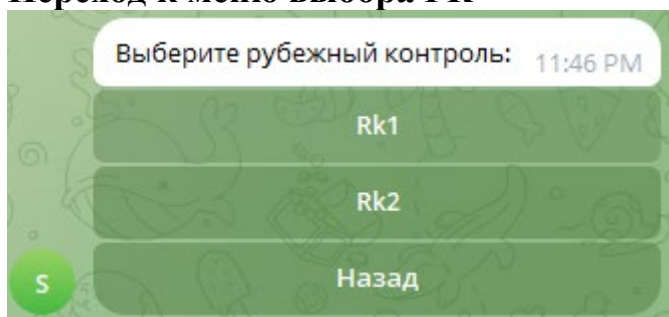


Описание лабораторных





Переход к меню выбора РК



Выбор РК

