

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

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

Курс «Парадигмы и конструкции языков программирования»

**Отчет по лабораторной работе №6
«Разработка простого бота для Telegram с использованием языка Python.»**

Выполнил:
студент группы РТ5-31Б:
Чиканчи В. А.

Подпись и дата:

Проверил:
преподаватель кафедры ИУ5
Гапанюк Ю. Е.

Подпись и дата:

Москва, 2024 г.

Постановка задачи

Разработайте простого бота для Telegram.

Текст программы

Файл «my_bot.py»:

```
from aiogram import Bot, Dispatcher, types
from aiogram.filters import Command
from aiogram.types import Message
import json
import os

BOT_TOKEN = '7882305908:AAFuWhjoT9t0KTCNLLnE6AJJjR4Z4nxU3cE'
# Создаем объекты бота и диспетчера
bot = Bot(token=BOT_TOKEN)
dp = Dispatcher()

battle_servers = {}

with open(os.path.join("users_settings.json"), "r", encoding="utf-8") as file:
    users_settings = json.load(file)

def users_settings_change():
    with open(os.path.join("users_settings.json"), "w") as file:
        json.dump(users_settings, file)

def create_battle_server(player1, player2):
    for i in range(10):
        if not i in battle_servers:
            battle_servers[i] = {
                "player1": player1,
                "player2": player2,
                "number_player1": -1,
                "number_player2": -1,
                "ans_player1": -1,
                "ans_player2": -1,
            }
            users_settings[player1]["battle_server_id"] = i
            users_settings[player2]["battle_server_id"] = i

def create_buttons():
    # markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    # btn1 = types.KeyboardButton("Поиск противника")
    # btn2 = types.KeyboardButton("Статус")
    markup = [
        types.KeyboardButton(text="Поиск противника"),
        types.KeyboardButton(text="Статус")]
    markup = types.ReplyKeyboardMarkup(keyboard=markup, resize_keyboard=True)
```

```

return markup

async def battling_players(player: str, text: str, id_server: int):

    if text.isdigit():
        if player == battle_servers[id_server]["player1"]:
            number_player = "number_player1"
            ans_player = "ans_player1"

            enemy = battle_servers[id_server]["player2"]
            number_enemy = "number_player2"
            ans_enemy = "ans_player2"
        else:
            number_player = "number_player2"
            ans_player = "ans_player2"

            enemy = battle_servers[id_server]["player1"]
            number_enemy = "number_player1"
            ans_enemy = "ans_player1"

        ans = int(text)

        if 0<=ans<=10:

            if battle_servers[id_server][number_player] == -1:
                if battle_servers[id_server][number_enemy] == -1:
                    await bot.send_message(player, "Ожидаем число вашего
противника.")
                else:
                    await bot.send_message(player, "Ваш противник уже загадал
число. Игра началась.")
                    await bot.send_message(enemy, "Игра началась.")
                    battle_servers[id_server][number_player] = ans

            elif battle_servers[id_server][number_enemy] == -1:
                await bot.send_message(player, "Противник ещё не загадал число.")

            elif battle_servers[id_server][ans_player] == -1:
                if battle_servers[id_server][ans_enemy] == -1:
                    await bot.send_message(player, "Ожидаем ответа противника.")

                battle_servers[id_server][ans_player] = ans
                if battle_servers[id_server][ans_player] != -1 and
battle_servers[id_server][ans_enemy] != -1:
                    if battle_servers[id_server][ans_player] ==
battle_servers[id_server][number_enemy] and battle_servers[id_server][ans_enemy]
== battle_servers[id_server][number_player]:
                        await bot.send_message(player, "Вы оба победили.",
reply_markup=create_buttons())

```

```

        await bot.send_message(enemy, "Вы оба победили.",
reply_markup=create_buttons())

        users_settings[player]["wins"]+=1
        users_settings[enemy]["wins"]+=1

        users_settings[player]["status"]="registrated"
        users_settings[enemy]["stasus"]="registrated"

        del battle_servers[id_server]
        users_settings_change()

    elif battle_servers[id_server][ans_player] ==
battle_servers[id_server][number_enemy]:
        await bot.send_message(player, "Вы победили.",
reply_markup=create_buttons())
        await bot.send_message(enemy, "Вы проиграли.",
reply_markup=create_buttons())

        users_settings[player]["wins"]+=1
        users_settings[enemy]["loses"]+=1

        users_settings[player]["status"]="registrated"
        users_settings[enemy]["stasus"]="registrated"

        users_settings_change()
        del battle_servers[id_server]

    elif battle_servers[id_server][ans_enemy] ==
battle_servers[id_server][number_player]:
        await bot.send_message(player, "Вы проиграли.",
reply_markup=create_buttons())
        await bot.send_message(enemy, "Вы победили.",
reply_markup=create_buttons())

        users_settings[player]["loses"]+=1
        users_settings[enemy]["wins"]+=1

        users_settings[player]["status"]="registrated"
        users_settings[enemy]["stasus"]="registrated"

        del battle_servers[id_server]
        users_settings_change()

    else:
        await bot.send_message(player, "Вы оба не угадали. Игра
продолжается.")
        await bot.send_message(enemy, "Вы оба не угадали. Игра
продолжается.")

        battle_servers[id_server][ans_enemy] = -1

```

```

        battle_servers[id_server][ans_player] = -1

    else:
        await bot.send_message(player, "Число должно быть от 0 до 10.")
    else:
        await bot.send_message(player, "Это не число.")

# Этот хэндлер будет срабатывать на команду "/start"
@dp.message(Command(commands=["start"]))
async def process_start_command(message: Message):

    if not str(message.from_user.id) in users_settings:
        users_settings[str(message.from_user.id)] = {
            "id": str(message.from_user.id),
            "name": "",
            "status": "registering",
            "battle_server_id": -1,
            "wins": 0,
            "loses": 0
        }
        await message.answer(f"Добрый день.\nВведите своё имя.")
    else:
        if users_settings[str(message.from_user.id)]["status"] == "battling":
            return
        name = users_settings[str(message.from_user.id)]["name"]
        await message.answer(f"Привет, {name}", reply_markup=create_buttons())

@dp.message(Command(commands=["battle"]))
async def process_battle_command(message: Message):
    if str(message.from_user.id) in users_settings:
        if users_settings[str(message.from_user.id)]["status"] == "battling":
            return
        await message.answer("Начинается поиск. Чтобы отменить, напишите /cancel.")

    for user in users_settings:

        if users_settings[user]["status"] == "searching":
            name1 = users_settings[str(message.from_user.id)]["name"]
            name2 = users_settings[user]["name"]

            await message.answer(f"Ваш противник: {name2}. Загадайте число")
            await bot.send_message(users_settings[user]["id"], f"Ваш
противник: {name1}. Загадайте число")

            create_battle_server(str(user), str(message.from_user.id))
            users_settings[str(message.from_user.id)]["status"] = "battling"
            users_settings[user]["status"] = "battling"

```

```

        break

    else:
        users_settings[str(message.from_user.id)][ "status" ] = "searching"

    else:
        await message.answer("Сначала напишите /start, чтобы пройти регистрацию.")

@dp.message(Command(commands=["cancel"]))
async def process_cancel_command(message: Message):
    if str(message.from_user.id) in users_settings:
        if users_settings[str(message.from_user.id)][ "status" ] == "searching":
            users_settings[str(message.from_user.id)][ "status" ] = "registered"
            await message.answer("Поиск отменён.", reply_markup=create_buttons())

@dp.message(Command(commands=["status"]))
async def process_status_command(message: Message):
    if str(message.from_user.id) in users_settings:
        if users_settings[str(message.from_user.id)][ "status" ] == "battling":
            return
        await message.answer(
            'Вот ваш статус:\n'
            f'побед: {users_settings[str(message.from_user.id)][ "wins" ]}\n'
            f'поражений {users_settings[str(message.from_user.id)][ "loses" ]}',
            reply_markup=create_buttons()
        )
    else:
        await message.answer("Сначала напишите /start, чтобы пройти регистрацию.")

# Этот хэндлер будет срабатывать на команду "/help"
@dp.message(Command(commands=['help']))
async def process_help_command(message: Message):
    markup = create_buttons()
    await message.answer(
        "Нажмите на кнопку (Поиск противника), чтобы начать игру\n",
        "Игроки должны выбрать число, которое хотят загадать. Потом они начинают пытаться отгадать число противника.", reply_markup=markup
    )

# Этот хэндлер будет срабатывать на любые ваши текстовые сообщения,
# кроме команд "/start" и "/help"
@dp.message()
async def send_echo(message: Message):
    try:
        if users_settings[str(message.from_user.id)][ "status" ] == "registering":
            users_settings[str(message.from_user.id)][ "name" ] = message.text

```

```

        users_settings[str(message.from_user.id)][ "status" ] = "registered"

        markup = create_buttons()

        await message.reply(text="Регистрация завершена",
reply_markup=markup)
        users_settings_change()
    except:
        await message.reply(text="Напишите /start, чтобы пройти регистрацию")
        return

    if users_settings[str(message.from_user.id)][ "status" ] == "battling":
        id_server = users_settings[str(message.from_user.id)][ "battle_server_id" ]
        await battling_players(str(message.from_user.id), message.text,
id_server)
        #print(battle_servers[id_server])
        #print(users_settings)

    elif message.text == "Поиск противника" and
users_settings[str(message.from_user.id)][ "status" ] != "battling":
        await process_battle_command(message)
    elif message.text == "Статус" and
users_settings[str(message.from_user.id)][ "status" ] != "battling":

        await process_status_command(message)

    else:
        if users_settings[str(message.from_user.id)][ "status" ] == "battling":
            return
        print(users_settings[str(message.from_user.id)][ "status" ])
        name = users_settings[str(message.from_user.id)][ "name" ]
        markup = create_buttons()

        await message.reply(text=f"{name}: {message.text}", reply_markup=markup)

```

Файл «main.py»:

```

from my_bot import *

if __name__ == '__main__':
    dp.run_polling(bot)

```

Экранные формы с примерами выполнения
программы

/start 13:17 ✓

Привет, nbitr 13:17

Статус 13:17 ✓

Вот ваш статус:
побед: 0
поражений 0 13:17

Поиск противника 13:17 ✓

Начинается поиск. Чтобы отменить, напишите /cansel. 13:17

/cansel 13:17 ✓

Поиск отменён. 13:17