

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

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»  
Отчет по лабораторной работе №5-6

Выполнил:  
студент группы ИУ5-31Б  
Коваленко  
Алексей Викторович

Подпись: \_\_\_\_\_

Дата: \_\_\_\_\_

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Юрий  
Евгеньевич

Подпись: \_\_\_\_\_

Дата: \_\_\_\_\_

Москва, 2021 г.

# **Лабораторная работа №5-6**

## **Описание задания**

### **Задание:**

1. Разработайте простого бота для Discord.
2. Разработайте бота для Discord. Бот должен реализовывать конечный автомат из трех состояний.

# Текст программы

Файл bot.py

```
import discord
import sqlite3
import time
from discord.ext import commands
from setting import settings

intents = discord.Intents(messages=True, guilds=True, members = True)
intents.reactions = True

TOKEN = settings['token']
client = commands.Bot(command_prefix='&', intents=intents)
client.remove_command('help')

connection = sqlite3.connect('oracle.db')
cursor = connection.cursor()

@client.event
async def on_ready():

    cursor.execute("""CREATE TABLE IF NOT EXISTS socrating (
        id INT,
        rating INT
    )""")
    connection.commit()

    cursor.execute("""CREATE TABLE IF NOT EXISTS warnings (
        user_id INT,
        admin_id INT,
        descr TEXT,
        amount INT,
        warntype TEXT,
        id INT
    )""")

    cursor.execute("""CREATE TABLE IF NOT EXISTS modcase (
        case_id INT,
        user_id INT,
        admin_id INT,
        descr TEXT
    )""")

    connection.commit()

    for guild in client.guilds:
        for member in guild.members:
            if cursor.execute(f"SELECT id FROM socrating WHERE id = {member.id}").fetchone() is None:
                cursor.execute(f"INSERT INTO socrating
VALUES ({member.id}, 30)")
            else:
                pass
    connection.commit()
    print('Join done')
    weekly()
```

```

@client.event
async def on_memeber_join(member):
    if cursor.execute(f"SELECT id FROM socrating WHERE id =
{member.id}").fetchone() is None:
        cursor.execute(f"INSERT INTO socrating
VALUES({member.id},30,{member})")
        connection.commit()
    else:
        pass

@client.command(aliases = ['rating'])
async def __rating(ctx, member: discord.Member=None):
    if member == None:
        await ctx.send(embed=discord.Embed(
            description = f""""Social rating of **{ctx.author}** is
**{cursor.execute(f"SELECT rating FROM socrating WHERE id =
{ctx.author.id}").fetchone()[0]** points"""""
        ))
    else:
        await ctx.send(embed=discord.Embed(
            description = f""""Social rating of **{member}** is
**{cursor.execute(f"SELECT rating FROM socrating WHERE id =
{member.id}").fetchone()[0]** points"""""
        ))

@client.command(aliases = ['addrating','add'])
@commands.has_any_role(*settings['admins'])
async def __addrating(ctx,member: discord.Member = None, amount : int=None):
    print('add triggered')
    if member==None:
        await ctx.send(embed=discord.Embed(
            description = "Please enter user, for who you want to manage
rating"
        ))
    else:
        if amount==None:
            await ctx.send(embed=discord.Embed(
                description = "Please enter amount of rating, which you want
to add"
            ))
        else:
            cursor.execute("UPDATE socrating SET rating = rating +{} WHERE id
= {}".format(amount,member.id))
            connection.commit()

            await ctx.message.add_reaction('✔')
            await notification(member,ctx)
            await cap_check(member)

@client.command(aliases = ['decreaserating','decrease','minus'])
@commands.has_any_role(*settings['admins'])
async def __decrating(ctx,member: discord.Member = None, amount : int=None):
    if member==None:
        await ctx.send(embed=discord.Embed(
            description = "Please enter user, for who you want to manage
rating"
        ))

```

```

        else:
            if amount==None:
                await ctx.send(embed=discord.Embed(
                    description = "Please enter amount of rating, which you want
to decrease"
                ))
            else:
                cursor.execute("UPDATE socrating SET rating = rating - {} WHERE
id = {}".format(amount,member.id))
                connection.commit()
                await ctx.message.add_reaction('✔')
                await notification(member,ctx)
                await cap_check(member)

@client.event
async def on_command_error(ctx, error):
    if isinstance(error, commands.MissingAnyRole):
        await ctx.send("*bonk* You have no permissions here")

async def weekly():
    sleep(604800)
    for guild in client.guilds:
        for member in guild.members:
            cursor.execute("UPDATE socrating SET rating = rating + 5 WHERE id
= {}".format(member.id))
            cap_check(member)
            notification(member)

    s
async def cap_check(member): #our current cap is 100
    amount= cursor.execute("SELECT rating FROM socrating WHERE id =
{}".format(member.id)).fetchone()[0]
    if amount>100:
        print('decrease done')
        cursor.execute("UPDATE socrating SET rating = 100 WHERE id =
{}".format(member.id))
        connection.commit()

async def notification(member,ctx):
    amount= cursor.execute("SELECT rating FROM socrating WHERE id =
{}".format(member.id)).fetchone()[0]
    if amount<25:
        print('notif done')
        logs = client.get_channel(settings['logs'])
        await logs.send(embed=discord.Embed(
            description = f""""**{member}** reached low level of reputation.
Current points is **{cursor.execute(f"SELECT rating FROM socrating WHERE id =
{ctx.author.id}").fetchone()[0]**.""")
        ))

@client.command(aliases = ['warn'])
@commands.has_any_role(*settings['admins'])
async def __warn(ctx,member: discord.Member = None, amount : int=None,
description: str =None):
    if member==None:
        await ctx.send(embed=discord.Embed(
            description = "Please enter user, for who you want to warn"
        ))
    else:
        if amount==None:
            await ctx.send(embed=discord.Embed(

```

```

        description = "Please enter amount of rating, which you want
to decrease"
    ))
    else:
        cursor.execute("UPDATE socrating SET rating = rating - {} WHERE
id = {}".format(amount, member.id))
        connection.commit()
        if amount < 10:
            warntype = 'lightwarning'
        elif amount >= 10 and amount < 25:
            warntype = 'warning'
        else:
            warntype = 'hardwarning'
        warn_id = cursor.execute(f"SELECT MAX(id) FROM
warnings").fetchone()[0]
        print(warn_id)
        cursor.execute(f"INSERT INTO warnings
VALUES({member.id},{ctx.author.id},{description},{amount},{warntype},{war
n_id+1})")
        connection.commit()
        settings['warn'] = warn_id
        connection.commit()
        await ctx.message.add_reaction('✔')
        await notification(member, ctx)
        await cap_check(member)

@client.command(aliases = ['unwarn'])
@commands.has_any_role(*settings['admins'])
async def __unwarn(ctx, case_id: int = None):
    if case_id == None:
        await ctx.send(embed=discord.Embed(
            description = "Please enter id of warning, which you want to
delete"
        ))
    else:
        amount = cursor.execute(f"SELECT amount FROM warnings WHERE id =
{case_id}").fetchone()[0]
        member = cursor.execute(f"SELECT user_id FROM warnings WHERE id =
{case_id}").fetchone()[0]
        cursor.execute("UPDATE socrating SET rating = rating + {} WHERE id =
{}".format(amount, member))
        cursor.execute(f"DELETE FROM warnings WHERE id = {case_id}")
        connection.commit()
        await ctx.message.add_reaction('✔')
        await notification(member, ctx)
        await cap_check(member)

@client.command(aliases = ['warnings'])
async def __warning(ctx, member: discord.Member = None):
    if member == None:
        await ctx.send(embed=discord.Embed(
            description = "Please enter user"
        ))
    else:
        amount = cursor.execute(f"SELECT COUNT(*) FROM warnings WHERE user_id
= {member.id}").fetchone()[0]
        res = cursor.execute(f"SELECT * FROM warnings WHERE user_id =
{member.id}").fetchmany(amount)
        desc = f"Warnings of {member} \n"
        for i in range(0, amount):
            desc += f"Case {res[i][5]} of {res[i][4]}: User -
{client.get_user(res[i][0])}, Admin -
{client.get_user(res[i][1])}, \nDescription: {res[i][2]}, Deacreated for

```

```

{res[i][3]}\n\n"
    await ctx.send(
        embed=discord.Embed(
            description = desc
        ))

#future undates 1
@client.command(aliases = ['help'])
async def __1warning(ctx,member: discord.Member = None, amount : int=None):
    pass
@client.command(aliases = ['admins'])
async def __2warning(ctx,member: discord.Member = None, amount : int=None):
    pass
@client.command(aliases = ['mods'])
async def __3warning(ctx,member: discord.Member = None, amount : int=None):
    pass

#future undates2
@client.command(aliases = ['mute'])
@commands.has_any_role(*settings['admins'])
async def __4warning(ctx,member: discord.Member = None, amount : int=None):
    pass
@client.command(aliases = ['unmute'])
@commands.has_any_role(*settings['admins'])
async def __5warning(ctx,member: discord.Member = None, amount : int=None):
    pass

#future undates3
@client.command(aliases = ['ban'])
@commands.has_any_role(*settings['admins'])
async def __6warning(ctx,member: discord.Member = None, amount : int=None):
    pass
@client.command(aliases = ['unban'])
@commands.has_any_role(*settings['admins'])
async def __7warning(ctx,member: discord.Member = None, amount : int=None):
    pass
@client.command(aliases = ['kick'])
@commands.has_any_role(*settings['admins'])
async def __8warning(ctx,member: discord.Member = None, amount : int=None):
    pass


#only lab thing. delete after lab
global state
state = 0
global joke_line
joke_line=[]

@client.event
async def on_message(message):
    global state
    if message.author != client.user:
        if state !=0:
            chan = client.get_channel(920072025303830599)

```

```

        if message.content == ' reset':
            await reset()
            await chan.send(embed=discord.Embed(
                description = 'Module reseted'))
        elif message.content == 'exit':
            await exit()
            await chan.send(embed=discord.Embed(
                description = 'Module off'))
        elif state == 1 :
            if message.content == 'back':
                await reset()
                await chan.send(embed=discord.Embed(
                    description = 'We are back to previous stage'))
            else:
                joke_line.append(message.content)
                state+=1
                await chan.send(embed=discord.Embed(
                    description = 'Enter second word to joke'))
        elif state ==2:
            if message.content == 'back':
                state-=1
                joke_line.pop()
                await chan.send(embed=discord.Embed(
                    description = 'We are back to previous stage'))
            else:
                joke_line.append(message.content)
                state+=1
                await chan.send(embed=discord.Embed(
                    description = 'Enter third word to joke'))
        elif state ==3:
            if message.content == 'back':
                state-=1
                joke_line.pop()
                await chan.send(embed=discord.Embed(
                    description = 'We are back to previous stage'))
            else:
                joke_line.append(message.content)
                state+=1
                await chan.send(embed=discord.Embed(
                    description = 'Do you want to see result of your
work?(Yes/No)'))
        elif state ==4:
            if message.content.lower()=='yes':
                await chan.send(embed=discord.Embed(
                    description = f'Идет {joke_line[0]} по
{joke_line[1]}, видит, {joke_line[2]} горит. Сел в нее и сторел.'))
                await reset()
            elif message.content.lower()=='no':
                await chan.send(embed=discord.Embed(
                    description = 'Im very sad to her that'))
                await reset()
            else:
                await chan.send(embed=discord.Embed(
                    description = 'I SAID YES OR NO!! moron'))
        await client.process_commands(message)

@client.command(aliases = ['start6'])
async def __start6(ctx):
    global state
    state = 1
    await ctx.send(embed=discord.Embed(
        description = 'Stupid joke module activated. Enter first word, whcih
we will add to our joke'

```



```
    ))

    async def reset():
        global state
        state = 1
        global joke_line
        joke_line = []

    async def exit():
        global state
        state = 0
        global joke_line
        joke_line = []

client.run(TOKEN)
```

## Экранные формы с примерами выполнения программы





Holy Rem Crusader Сегодня, в 4:14

&add @Тюлеша младший 1000



1

&rating @Тюлеша младший



Oracle БОТ Сегодня, в 4:14

Social rating of Тюлеша младший#5622 is 100 points



Holy Rem Crusader Сегодня, в 4:29

&warn @Holy Rem Crusader 10 baka



1

&warnings @Holy Rem Crusader



Oracle БОТ Сегодня, в 4:29

Warnings of Holy Rem Crusader#6288

Case 1 of warning: User - Holy Rem Crusader#6288, Admin - Holy Rem Crusader#6288,

Description:baka, Deacreated for 10



Holy Rem Crusader Сегодня, в 4:29

&warn @Тюлеша младший 20 alt



1

&warnings @Тюлеша младший



Oracle БОТ Сегодня, в 4:30

Warnings of Тюлеша младший#5622

Case 2 of warning: User - Тюлеша младший#5622, Admin - Holy Rem Crusader#6288,

Description:alt, Deacreated for 20





**Holy Rem Crusader** Сегодня, в 4:30  
&start6



**Oracle БОТ** Сегодня, в 4:30

Stipid joke module activated. Enter first word, whcih we will add to our joke



**Holy Rem Crusader** Сегодня, в 4:30  
Медведь



**Oracle БОТ** Сегодня, в 4:30

Enter second word to joke



**Holy Rem Crusader** Сегодня, в 4:30  
ЛЕС



**Oracle БОТ** Сегодня, в 4:30

Enter third word to joke



**Holy Rem Crusader** Сегодня, в 4:30  
Машина



**Oracle БОТ** Сегодня, в 4:30

Do you want to see result of your work?(Yes/No)



**Holy Rem Crusader** Сегодня, в 4:30  
Yes



**Oracle БОТ** Сегодня, в 4:30

Идет Медведь по ЛЕС, видит, Машина горит. Сел в нее и сгорел.



Holy Rem Crusader Сегодня, в 4:30  
Yes



Oracle БОТ Сегодня, в 4:30

Идет Медведь по ЛЕс, видит, Машина горит. Сел в нее и сгорел.



Holy Rem Crusader Сегодня, в 4:31  
Двачер



Oracle БОТ Сегодня, в 4:31

Enter second word to joke



Holy Rem Crusader Сегодня, в 4:31  
борда



Oracle БОТ Сегодня, в 4:31

Enter third word to joke



Holy Rem Crusader Сегодня, в 4:31  
а



Oracle БОТ Сегодня, в 4:31

Do you want to see result of your work?(Yes/No)



Holy Rem Crusader Сегодня, в 4:31  
No



Oracle БОТ Сегодня, в 4:31

Im very sad to her that





Holy Rem Crusader Сегодня, в 4:31

No



Oracle BOT Сегодня, в 4:31

Im very sad to her that



Holy Rem Crusader Сегодня, в 4:31

1



Oracle BOT Сегодня, в 4:31

Enter second word to joke



Holy Rem Crusader Сегодня, в 4:31

2



Oracle BOT Сегодня, в 4:31

Enter third word to joke



Holy Rem Crusader Сегодня, в 4:31

3



Oracle BOT Сегодня, в 4:31

Do you want to see result of your work?(Yes/No)



Holy Rem Crusader Сегодня, в 4:31

е



Oracle BOT Сегодня, в 4:31

I SAID YES OR NO!! moron



**Holy Rem Crusader** Сегодня, в 4:32  
exit



**Oracle** БОТ Сегодня, в 4:32

Module off