



*«Московский государственный технический университет  
имени Н.Э. Баумана»*

*(МГТУ им. Н.Э. Баумана)*

---

ФАКУЛЬТЕТ

Информатика и системы управления

КАФЕДРА

Системы обработки информации и управления

## **О т ч ё т  п о  л а б о р а т о р н о й  р а б о т е**

### **п о  к у р с у**

### **« Р а з р а б о т к а  и н т е р н е т - п р и л о ж е н и й »**

### **Python - классы**

Исполнитель: студентка группы **РТ5-51**

**Галичий Д.А.**

Преподаватель: **Гапанюк Ю.Е.**

**Цель работы:** ознакомление с модулями и ООП в Python, освоение работы с сетью, создание набора классов для реализации работы с VK API.

**Задание:** построить гистограмму распределения возрастов друзей пользователя, username которого является входным параметром программы.

Содержание файла «base\_client.py»:

```
class BaseClient:
    # URL vk api
    BASE_URL = None
    # метод vk api
    method = None
    # GET, POST, ...
    http_method = None

    # Получение GET параметров запроса
    def get_params(self):
        return None

    # Получение данных POST запроса
    def get_json(self):
        return None

    # Получение HTTP заголовков
    def get_headers(self):
        return None

    # Склейка url
    def generate_url(self, method):
        return '{0}{1}'.format(self.BASE_URL, method)

    # Отправка запроса к VK API
    def _get_data(self, method, http_method):
        response = None

        # todo ВЫПОЛНИТЬ запрос

        return self.response_handler(response)

    # Обработка ответа от VK API
    def response_handler(self, response):
        return response

    # Запуск клиента
    def execute(self):
        return self._get_data(
            self.method,
```

```
        http_method=self.http_method
    )
```

Содержание файла «get\_id\_from\_username.py»:

```
from base_client import BaseClient
import requests
import json

class GetIDFromUsername(BaseClient):
    # URL vk api
    BASE_URL = 'https://api.vk.com/method/'
    # GET
    http_method = 'users.get'

    # Method, which is called when the exemplar is creating
    def __init__(self, uname):
        self.name = uname

    # Taking of GET-parameters of the request
    def get_params(self):
        return 'user_ids=' + self.name

    # Sending of the request to VK API
    def _get_data(self, method, http_method):
        response =
requests.get(self.generate_url(self.http_method) + '?' +
self.get_params())
        return self.response_handler(response)

    # Handling of the response from VK API
    def response_handler(self, response):
        try:
            str = json.loads(response.text)
            return str['response'][0]['uid']
        except:
            raise Exception("The response from VK API couldn't
been handled.")
```

Содержание файла «get\_list\_of\_friends.py»:

```
from base_client import BaseClient
import requests
import json
from datetime import datetime

class GetFriends(BaseClient):
    # URL vk api
    BASE_URL = 'https://api.vk.com/method/'
```

```

# GET
http_method = 'friends.get'

# Method, which is called when the exemplar is creating
def __init__(self, uid):
    self.uid = uid

# Taking of GET-parameters of the request
def get_params(self):
    return 'user_id=' + str(self.uid) + '&fields=bdate'

# Sending of the request to VK API
def _get_data(self, method, http_method):
    response =
requests.get(self.generate_url(self.http_method) + '?' +
self.get_params())
    return self.response_handler(response)

# Handling of the response from VK API
def response_handler(self, response):
    try:
        str = json.loads(response.text)
        friends = str.get('response')
        ages_amount = {}
        undef_age = []

        for friend in friends:
            try:
                bdate = friend.get('bdate')
                bdate = datetime.strptime(bdate,
'%d.%m.%Y')

            except:
                undef_age.append('#')
                continue

            ndate = datetime.now()
            age = (int((ndate - bdate).days))//365

            if (age in ages_amount):
                ages_amount[age] += '#'
            else:
                ages_amount[age] = '#'

        return ages_amount, undef_age

    except:
        raise Exception("The response from VK API for
friends couldn't been handled.")

```

Содержание файла «main.py»:

```

from get_id_from_username import GetIDFromUsername
from get_list_of_friends import GetFriends
import matplotlib.pyplot as plt

username = input('Username: ')
user = GetIDFromUsername(username)
uid = user.execute()

user_friends = GetFriends(uid)
statistics, undef_age = user_friends.execute()

X = []
Y = []
for k in sorted(statistics.keys()):
    print(k, statistics[k])
    X.append(k)
    Y.append(len(statistics[k]))
print ('- ', end='')
for el in undef_age:
    print(el, end='')

fig, ax = plt.subplots()
plt.bar(X, Y, align='center')
ax.set_xlabel('Age')
ax.set_ylabel('Amount of friends')
ax.set_title(r'Histogram of a spreading of ages')
plt.show()

```

Результат работы программы:

```

Run main
C:\Users\Дарья\AppData\Local\Programs\Python\Python3
Username: 111111
23 #
27 ###
28 #####
29 ###
31 ##
32 #
34 ###
- #####
Process finished with exit code 0

```

